



UNIVERSIDADE ESTADUAL PAULISTA
“JÚLIO DE MESQUITA FILHO”
Instituto de Ciência e Tecnologia
Câmpus de Sorocaba

GIOVANNI DOTTA BRIGANTI

Sistema de gerenciamento de processos na área de saúde –
Auto check-in para procedimentos

Sorocaba

2024

GIOVANNI DOTTA BRIGANTI

SISTEMA DE GERENCIAMENTO DE PROCESSOS NA ÁREA
DE SAÚDE - AUTO *CHECK-IN* PARA PROCEDIMENTOS

Trabalho de Conclusão de Curso apresentado ao Instituto de Ciência e Tecnologia de Sorocaba, Universidade Estadual Paulista (UNESP), como parte dos requisitos para obtenção do grau de Bacharel em Engenharia de Controle e Automação.

Orientador: Prof. Dr. Galdenoro Botura Junior

Sorocaba

2024

B854s

Briganti, Giovanni Dotta

Sistema de gerenciamento de processos na área de saúde : Auto check-in para procedimentos / Giovanni Dotta Briganti. -- Sorocaba, 2024

102 p. : il., tabs., fotos

Trabalho de conclusão de curso (Bacharelado - Engenharia de Controle e Automação) - Universidade Estadual Paulista (UNESP), Instituto de Ciência e Tecnologia, Sorocaba

Orientador: Galdenoro Botura Junior

1. Saúde – Administração. 2. Gerenciamento de recursos de informação. 3. Interface de programação de aplicativos (software de computador). 4. JavaScript (Linguagem de programação de computador). 5. MySQL (Recurso eletrônico). I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca da Universidade Estadual Paulista (UNESP), Instituto de Ciência e Tecnologia, Sorocaba. Dados fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

SISTEMA DE GERENCIAMENTO DE PROCESSOS NA ÁREA DE SAÚDE
- AUTO *CHECK-IN* PARA PROCEDIMENTOS

GIOVANNI DOTTA BRIGANTI

ESTE PROJETO FINAL DE CURSO FOI JULGADO ADEQUADO
COMO PARTE DO REQUISITO PARA A OBTENÇÃO DO GRAU DE
BACHAREL EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Prof. Dr. Everson Martins
Coordenador

BANCA EXAMINADORA:

Prof. Dr. GALDENORO BOTURA JUNIOR
Orientador/UNESP – Câmpus de Sorocaba

Prof. Dr. Eduardo Verri Liberado
UNESP – Câmpus de Sorocaba

Prof. Dr. Leopoldo André Dutra Lusquino Filho
UNESP – Câmpus de Sorocaba

29 de maio de 2024

AGRADECIMENTOS

Começo os agradecimentos a minha família, em destaque meus pais Johnny Briganti e Jeane Dotta Briganti, nos quais me deram o suporte para estar traçando esta caminhada até o momento, além de confiar em meu potencial.

Agradeço ao Prof. Dr. Galdenoro Botura Junior pela orientação do projeto no qual guiou a uma alta qualidade de um trabalho completo.

Agradeço aos meus amigos Guilherme Doubek, Lucas Lima e Matheus Nunes que a faculdade me apresentou e levarei para a vida toda, além de ser eternamente grato por todo suporte ao longo dessa caminhada na universidade, principalmente neste trabalho. Destaco meus agradecimentos também ao Gabriel Siles e Daniel Francisco, dois companheiros de curso que respeito e desejo muito sucesso, que mesmo distanciando-se no final da graduação foram extremamente importantes durante a minha estadia na universidade.

Agradeço também ao Henrique Hayashi e Pedro Pecora, amigos de longa data que levo para vida e continuarei levando, com quem sempre pude contar e os mesmo puderam e poderão contar comigo para o que der e vier, e que sem eles este trabalho não seria possível.

Agradeço aos meus amigos Aurea Pucci e Vinícius Pucci, que sem eles a ideia desenvolvida neste trabalho não existiria.

E termino meus agradecimentos a todas as pessoas que fazem a UNESP, em especial o campus de Sorocaba, ser a universidade de peso que ela é. Que os esforços de todos foram e são impactantes em minha vida e tornaram possível a minha formação como engenheiro. Meu mais sincero e significativo: Muito Obrigado!

RESUMO

Agilidade, redução do contato físico e utilização otimizada de recursos não são pontos de melhoria exclusivos da área da saúde, contudo, a pandemia de COVID-19 evidenciou o quão cruciais podem ser tais características em um contexto hospitalar. Mesmo após uma crise sanitária global, a partir de entrevistas com profissionais atuando nas redes pública e privada da região metropolitana de Sorocaba, constatou-se que ainda há problemas de comunicação entre os funcionários e integração de recursos nos hospitais locais, ocasionalmente gerando atrasos, cancelamentos de procedimentos e insatisfação por parte dos pacientes. Diante disso, o presente trabalho propõe elaborar um Produto Viável Mínimo (MVP) de uma aplicação digital de auto *check-in* de pacientes e automação de encaminhamento de procedimentos médicos. Para o desenvolvimento de tal trabalho, utilizou-se o Figma para a definição do *layout* das páginas do sistema, o *software* dbdiagram.io para modelagem do banco de dados e as linguagens JavaScript e TypeScript, intermediadas pelas ferramentas ReactJs e NodeJs, respectivamente, para a construção do *Front-end* e *back-end* do sistema. Foi possível desenvolver a aplicação e disponibilizar na internet para que pudesse ser acessada por um *link*. A aplicação foi testada por diversas pessoas, incluindo algumas que trabalham na área da saúde, as quais disponibilizaram suas considerações a partir do preenchimento de um formulário. As respostas obtidas foram promissoras quanto à implementação do sistema, destacando as funcionalidades oferecidas, usabilidade e inovação. Porém foram ressaltados alguns pontos de melhorias e um problema isolado na automação do encaminhamento que precisariam ser resolvidos antes de disponibilizá-lo comercialmente. Com base nisso foi possível concluir que o sistema criado possui um potencial benefício a ser explorado para a área de saúde, a partir da adequação às sugestões de melhorias coletadas, e que demonstrou ser bem-sucedido na integração e gerenciamento de processos para a área de saúde, com enfoque no auto *check-in* para procedimentos médicos.

Palavras-chave: auto *check-in*; automação de processos; MVP; sistema de gerenciamento; área da saúde.

ABSTRACT

Agility, reduced physical contact and optimized resource utilization are not exclusive areas for improvement in healthcare; however, the COVID-19 pandemic has demonstrated how crucial such characteristics can be in medical area. Even after a global health crisis, interviews with professionals working in the public and private facilities of the Sorocaba metropolitan region revealed ongoing communication issues among staff and resource integration in local hospitals, occasionally leading to delays, procedure cancellations, and patient dissatisfaction. Therefore, this project proposes to develop a Minimum Viable Product (MVP) for a digital patient self-*check-in* and medical procedure automation application. For the development of this project, Figma was used to define the system's page *layout*, dbdiagram.io for database modeling, and JavaScript and TypeScript languages, facilitated by ReactJs and NodeJs tools, respectively, for building the system's *Front-end* and *back-end*. The application was developed and made available on the internet accessible via a *link*. It was tested by several individuals, including healthcare professionals, who provided *feedback* through a form. The responses were promising regarding system implementation, highlighting the functionalities, usability, and innovation. However, some areas for improvement were noted, including an isolated issue with the automation of procedures that would need resolution before commercial release. Based on this, it was concluded that the developed system holds potential benefits to be explored in healthcare, through adaptation to collected improvement suggestions, demonstrating success in integrating and managing processes within the healthcare field, with a focus on self-*check-in* for medical procedures.

Keywords: self-*check-in*; process automation; MVP; management system; health sector.

LISTA DE FIGURAS

Figura 1 - Gráfico de projeção de crescimento do mercado global de automação industrial em bilhões de dólares	13
Figura 2 - Diagrama que representa a relação de uma API com os clientes e o servidor	16
Figura 3 - Foto de totens de auto <i>check-in</i> da companhia aérea Iberia	18
Figura 4 - Fluxo idealizado para o usuário do tipo Admin.....	24
Figura 5 - Fluxo idealizado para o usuário do tipo Médico.....	25
Figura 6 - Fluxo idealizado para o usuário do tipo Enfermeiro.....	26
Figura 7 - Fluxo idealizado para o usuário do tipo Paciente	26
Figura 8 - Fluxo idealizado para o processo de <i>Check-in</i>	27
Figura 9 - Componentes visuais dos menus laterais construídos no Figma, sendo que o menu à direita é o que foi utilizado nas páginas Inicial e <i>Check-in</i> e o menu à esquerda é o que foi utilizado nas demais páginas	28
Figura 10 - Parte de Usuários presente no modelo do banco de dados construído no dbdiagram.io.....	31
Figura 11 - <i>Endpoint</i> de listagem de procedimento médicos e seu retorno no formato JSON feito no <i>software</i> Talend API Tester, utilizado para testar os <i>endpoints</i> disponibilizados pelo servidor	33
Figura 12 - Resultado da geração de arquivo de PDF para <i>download</i> , após confirmação do <i>check-in</i>	40
Figura 13 - Menu lateral construído para ser utilizado nas páginas de <i>Home</i> e <i>Check-in</i>	41
Figura 14 - Menu lateral fornecido para o tipo de usuário Enfermeiro e que também utilizado para os demais tipos de usuários.....	42
Figura 15 - Página Inicial com o menu contendo o logo do sistema e os <i>cards</i> que dão acesso às páginas do sistema.....	43
Figura 16 - Página de <i>Check-in</i> apresentando o estado inicial, com a instrução de inserir o CPF	44
Figura 17 - Página de <i>Check-in</i> com a apresentação do estado de erro, com a instrução que não há procedimento para o paciente	44
Figura 18 - Página de <i>Check-in</i> com procedimento encontrado na base de dados, contendo as informações do paciente, do procedimento e o botão de <i>Check-in</i>	45
Figura 19 - Página de <i>Check-in</i> com apresentação de mensagem de <i>check-in</i> bem-sucedido .	45

Figura 20 - Página de <i>Check-in</i> com apresentação da mensagem informando que o <i>check-in</i> já foi realizado	46
Figura 21 - Página de listagem de usuários na visão de um Admin.....	46
Figura 22 - Página de detalhe de um usuário do tipo Paciente.....	47
Figura 23 - Página de listagem de procedimentos na visão de um usuário do tipo Admin.....	48
Figura 24 - Página de detalhes do procedimento médico com enfoque nas Informações Gerais	48
Figura 25 - Página de detalhes do procedimento médico com enfoque nos Exames	49
Figura 26 - Comportamento de carregamento ao fazer o upload de arquivo	49
Figura 27 - Página de detalhes do procedimento médico com enfoque nas Consultas	50
Figura 28 - Página de detalhes do procedimento médico com enfoque no Histórico Médico .	50
Figura 29 - Página de detalhes do procedimento médico com enfoque no Prontuário do enfermeiro.....	51
Figura 30 - Página de detalhes do procedimento médico com enfoque no Prontuário do médico	51
Figura 31 - Página de detalhes do procedimento médico com enfoque na Prescrição Médica	52
Figura 32 - Página de encaminhamento com a definição de horário, os botões de ação (Pesquisar e Encaminhar) e a tabela dos procedimentos filtrados	53
Figura 33 - Página de encaminhamento em execução (após clicar no botão ENCAMINHAR) destacando os nomes a serem chamados para os respectivos procedimentos médicos.....	54
Figura 34 - Gráfico de avaliação dos tipos de usuários em que os participantes se enquadram	55
Figura 35 - Gráfico de avaliação da intuitividade e facilidade de uso do Fluxo de Admin	56
Figura 36 - Gráfico de avaliação da abrangência das funcionalidades do Fluxo de Admin	56
Figura 37 - Gráfico de avaliação da qualidade das funcionalidades do Fluxo de Admin	57
Figura 38 - Gráfico de avaliação da intuitividade e facilidade de uso do Fluxo de Médico	57
Figura 39 - Comentário referente à avaliação baixa da intuitividade e facilidade de uso do Fluxo de Médico	58
Figura 40 - Gráfico de avaliação da abrangência das funcionalidades do Fluxo de Médico ...	58
Figura 41 - Gráfico de avaliação da qualidade das funcionalidades do Fluxo de Médico	59
Figura 42 - Gráfico de avaliação da intuitividade e facilidade de uso do Fluxo de Enfermeiro	60
Figura 43 - Gráfico de avaliação da abrangência das funcionalidades do Fluxo de Enfermeiro	60

Figura 44 - Comentário referente à avaliação mediada da abrangência das funcionalidades do Fluxo de Enfermeiro	61
Figura 45 - Gráfico de avaliação da qualidade das funcionalidades do Fluxo de Enfermeiro .	61
Figura 46 - Gráfico de avaliação da intuitividade e facilidade de uso do Fluxo de Paciente...	62
Figura 47 - Gráfico de avaliação da abrangência das funcionalidades do Fluxo de Paciente..	62
Figura 48 - Comentário referente à avaliação baixa da abrangência das funcionalidades do Fluxo de Paciente	63
Figura 49 - Gráfico de avaliação da qualidade das funcionalidades do Fluxo de Paciente.....	63
Figura 50 - Gráfico da avaliação do grau de inovação do sistema	64
Figura 51 - Gráfico da avaliação do grau do impacto do sistema	64
Figura 52 - Gráfico da avaliação da eficiência e eficácia do sistema.....	65
Figura 53 - Gráfico da avaliação da satisfação dos usuários ao utilizar o sistema.....	66
Figura 54 - Gráfico da avaliação da aceitação do sistema.....	66

LISTA DE QUADROS

Quadro 1 - Definição e descrição das telas construídas no Figma	29
Quadro 2 - Definição das rotas construídas na aplicação <i>front-end</i>	30
Quadro 3 - Definição das variáveis que compõem a automação presente na página de encaminhamento	35
Quadro 4 - Relação das rotas de Usuários disponibilizadas pelo servidor	38
Quadro 5 - Relação das rotas de Procedimentos Médicos disponibilizadas pelo servidor.....	39
Quadro 6 - Relação das rotas de <i>Check-in</i> disponibilizadas pelo servidor.....	40
Quadro 7 - Relação das rotas de Encaminhamento disponibilizadas pelo servidor	40
Quadro 8 - Relação das rotas de Exames disponibilizadas pelo servidor	41

SUMÁRIO

1	INTRODUÇÃO.....	13
2	REVISÃO DE LITERATURA	15
2.1	Inovação.....	15
2.2	Sistemas automatizados para gerenciamento de dados.....	15
2.3	API (Interface de Programação de Aplicativos)	16
2.4	MVP (Produto Mínimo Viável)	17
2.5	Auto <i>check-in</i>	17
2.6	Tecnologias	19
2.6.1	<i>JavaScript</i>.....	19
2.6.2	<i>TypeScript</i>.....	19
2.6.3	<i>React.js</i>.....	20
2.6.4	<i>Node.js</i>	21
2.6.5	<i>MySQL</i>.....	21
3	METODOLOGIA	23
3.1	Apresentação dos requisitos.....	23
3.1.1	<i>Requisitos funcionais</i>.....	23
3.1.1.1	<i>Requisitos do fluxo Administrador</i>.....	24
3.1.1.2	<i>Requisitos do fluxo Médico</i>.....	24
3.1.1.3	<i>Requisitos do fluxo Enfermeiro</i>.....	25
3.1.1.4	<i>Requisitos do fluxo Paciente</i>	26
3.1.2	<i>Requisitos técnicos</i>	27
3.1.2.1	<i>Construção das telas do sistema CheckInMed</i>	28
3.1.2.2	<i>Modelagem do banco de dados relacional</i>	30
3.1.2.3	<i>Desenvolvimento das funcionalidades do servidor e comunicação com o banco de dados</i>	31
3.1.2.4	<i>Automação do processo de encaminhamento de pacientes</i>	34
3.1.2.5	<i>Publicação das aplicações</i>.....	35
3.1.2.6	<i>Validação do protótipo</i>	36
4	RESULTADOS E DISCUSSÕES	37
4.1	Construção e população do banco de dados	37
4.2	API fornecida pelo <i>back-end</i> do sistema	37
4.3	Páginas construídas para o sistema.....	41
4.3.1	<i>Menus laterais utilizados nas páginas do sistema</i>	41

4.3.2	<i>Página inicial (Home)</i>	42
4.3.3	<i>Página de check-in</i>	43
4.3.4	<i>Páginas de listagem e detalhes de usuário</i>	46
4.3.5	<i>Páginas de listagem e detalhes de procedimentos</i>	47
4.3.5.1	<i>Componente 1 – Informações Gerais</i>	48
4.3.5.2	<i>Componente 2 – Exames</i>	49
4.3.5.3	<i>Componente 3 – Consultas</i>	50
4.3.5.4	<i>Componente 4 – Histórico Médico</i>	50
4.3.5.5	<i>Componente 5 – Prontuário (Enfermeiro)</i>	51
4.3.5.6	<i>Componente 6 – Prontuário (Médico)</i>	51
4.3.5.7	<i>Componente 7 – Prescrição Médica</i>	52
4.3.6	<i>Página de encaminhamento</i>	53
4.4	Apresentação e discussão das respostas do formulário de validação do sistema...	54
4.4.1	<i>Caracterização dos participantes do processo de validação</i>	55
4.4.2	<i>Funcionalidade e usabilidade</i>	55
4.4.2.1	<i>Fluxo de Admin</i>	55
4.4.2.2	<i>Fluxo de Médico</i>	57
4.4.2.3	<i>Fluxo de Enfermeiro</i>	59
4.4.2.4	<i>Fluxo de Paciente</i>	62
4.4.3	<i>Inovação</i>	63
4.4.4	<i>Satisfação e aceitação</i>	65
5	CONCLUSÃO	68
	REFERÊNCIAS	70
	GLOSSÁRIO	74
	APÊNDICE A – Design das telas planejadas para o sistema CheckInMed modeladas no software Figma	77
	APÊNDICE B – Modelo do banco de dados proposto	84
	APÊNDICE C – Script de construção das tabelas do banco de dados MySQL gerado pelo software de modelagem dbdiagram.io para o sistema CheckInMed	85
	APÊNDICE D – Script de população das tabelas do banco de dados MySQL para o sistema CheckInMed	89
	APÊNDICE E – Documento PDF que representa a prescrição médica para impressão, gerada para o navegador	96
	APÊNDICE F – Link do deploy do protótipo	97
	APÊNDICE G – Seção de comentários de melhorias referente ao fluxo de Admin	98

APÊNDICE H – Seção de comentários de melhorias referente ao fluxo de Médico.....	99
APÊNDICE I – Seção de comentários de melhorias referente ao fluxo de Enfermeiro	100
APÊNDICE J – Seção de comentários de melhorias referente ao fluxo de Paciente	101
APÊNDICE K – Justificativas de aceitação do sistema CheckInMed.....	102

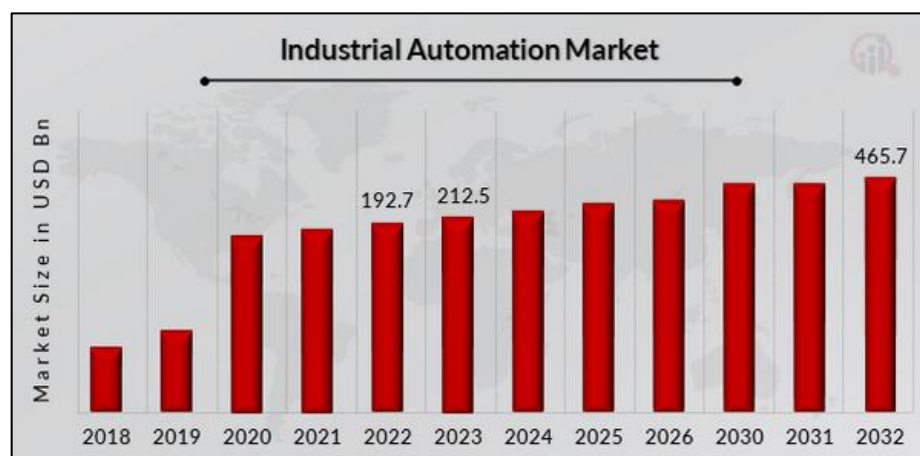
1 INTRODUÇÃO

O avanço acelerado da tecnologia tem sido marcado pelo surgimento e aprimoramento da automação, um fenômeno que redefine a forma como as sociedades organizam e executam suas atividades. Com a gradual introdução de sistemas autônomos em todo e qualquer ambiente onde há possibilidade de melhoria através das mais recentes aplicações desenvolvidas, incluindo até mesmo inteligência artificial, inúmeros setores econômicos têm obtido impactos significativos que moldam a dinâmica do mundo moderno (CHIMIELEWSKI; SILVA; LEITE, 2020).

Em primeiro lugar, a automação tem sido uma força impulsionadora da eficiência e produtividade. Máquinas e sistemas automatizados conseguem realizar tarefas complexas em um piscar de olhos, de maneira precisa e sem a necessidade de descanso, o que leva a um aumento substancial na produção de bens, na qualidade dos produtos e na eficiência na prestação de serviços (YANK SOLUTIONS, 2023). Além disso, a automação desempenha um papel crucial na segurança e na redução de riscos em ambientes de trabalho. Tarefas perigosas, monótonas ou que envolvem exposição a condições de risco podem ser transferidas para máquinas, protegendo a saúde e a segurança dos trabalhadores.

No gráfico da Figura 1 é possível observar a projeção de crescimento do mercado global de automação industrial, culminando em 465 bilhões de dólares em 2032. Destaca-se também que houve um crescimento expressivo do mesmo de 2019 para 2020 como efeito direto da crise global causada pelo COVID-19. Muitas organizações precisaram se adaptar a novos modelos de negócios devido às restrições impostas pelas medidas de supressão de contágio. A automação tornou-se uma ferramenta imprescindível para essas mudanças, permitindo a entrega de serviços de forma mais eficiente e flexível mesmo sob as dificuldades do período.

Figura 1 - Gráfico de projeção de crescimento do mercado global de automação industrial em bilhões de dólares



Fonte: Market Research Future (2023).

Sendo um dos setores mais diretamente impactados durante a pandemia, ocorreu uma considerável aceleração na adoção de soluções automatizadas na área da saúde. Com o objetivo de reduzir o contato humano e fortalecer a capacidade de resposta dos hospitais, houve o advento de diversas ferramentas novas ou melhoradas, como a robótica cirúrgica, aplicativos de convênios médicos e a telemedicina (CASTRO; LIMA; COSTA, 2020). Essas e muitas outras tecnologias estão possibilitando uma abordagem mais ágil e eficaz no atendimento, ao mesmo tempo em que otimizam os processos internos das instituições.

No entanto, ainda existem pontos de melhoria carentes de novas abordagens em ambientes hospitalares. A partir de relatos obtidos por meio de entrevistas com profissionais da área atuando nas redes pública e privada da região metropolitana de Sorocaba, constatou-se que há problemas de comunicação entre os funcionários, ocasionalmente gerando atrasos, cancelamentos de procedimentos e descontentamento por parte dos pacientes. A implementação de um sistema de autoatendimento e gerenciamento de agendamentos foi sugerida pelos mesmos como possível resposta para muitas destas ocorrências.

Por conseguinte, o objetivo deste trabalho é elaborar um Produto Viável Mínimo (MVP) de uma aplicação digital de auto *check-in* de pacientes e automação de encaminhamento de processos para os diversos departamentos em clínicas e/ou hospitais, visando a diminuição de falhas humanas, redução de carga de trabalho para recepções e o aumento da satisfação dos clientes, empregando as mais modernas tecnologias de desenvolvimento do mercado atual.

Para o desenvolvimento de tal projeto, o presente trabalho contou com conceitos relacionados a sistemas de gerenciamento de dados, sistemas de auto *check-in* em diferentes ambientes e tecnologias que foram empregadas para a construção do *software*. Além disso, foram definidos os requisitos que o sistema deve cumprir, sendo eles a construção das interfaces de usuário, a modelagem do banco de dados, as funcionalidades do sistema (como geração de PDFs e gerenciamento de arquivos com o Google Drive), uma automação para encaminhamento de pacientes, publicação do sistema na rede e validação por meio de formulário. Por fim, para os resultados obtidos, analisou-se o sistema por completo e as respostas do processo de validação.

2 REVISÃO DE LITERATURA

Nesta seção são apresentados os conceitos, tecnologias e recursos que nortearam a execução do presente trabalho, sendo divididos nas seguintes seções: inovação, sistemas automatizados para gerenciamento de dados, API (Interface de Programação de Aplicativos), MVP (Produto Mínimo Viável), Auto *Check-in* e tecnologias (Javascript, TypeScript, React.js, Node.js e MySQL).

2.1 Inovação

A inovação é a capacidade de introduzir algo novo ou significativamente melhorado em produtos, processos, serviços, modelos de negócios ou práticas organizacionais, visando gerar valor econômico e/ou social. Não se restringe apenas à criação de novas ideias, mas também à sua implementação bem-sucedida, resultando em benefícios tangíveis e impacto positivo. Existem diversos tipos de inovação, como a inovação de produto, que envolve a introdução de bens ou serviços novos ou aprimorados; a inovação de processo, que se concentra na implementação de novos métodos ou processos para melhorar a eficiência e reduzir custos; e a inovação organizacional, que se refere à introdução de novas práticas de gestão e estruturas organizacionais para melhorar o desempenho e a competitividade (MERCALDI et al., 2015).

A aplicação da inovação em um contexto real é essencial para o desenvolvimento e crescimento das organizações, permitindo-lhes se manter competitivas no mercado. Isso pode ser realizado através do desenvolvimento de novos produtos ou serviços que atendam às necessidades dos clientes de maneira única, da melhoria de processos internos para otimizar a eficiência operacional e reduzir custos. Além disso, a inovação organizacional promove novas estruturas e práticas de gestão que estimulam a criatividade e a adaptação às mudanças do mercado, enquanto a colaboração e parcerias com outras empresas ou instituições facilitam a troca de conhecimentos e recursos inovadores (GARCIA; BOTURA JUNIOR; SILVA, 2023).

2.2 Sistemas automatizados para gerenciamento de dados

Sistemas automatizados são conjuntos de dispositivos, processos e *softwares* que operam de forma autônoma ou semiautônoma para realizar tarefas específicas ou gerenciar processos complexos. Eles são projetados para executar operações de forma eficiente e precisa, muitas

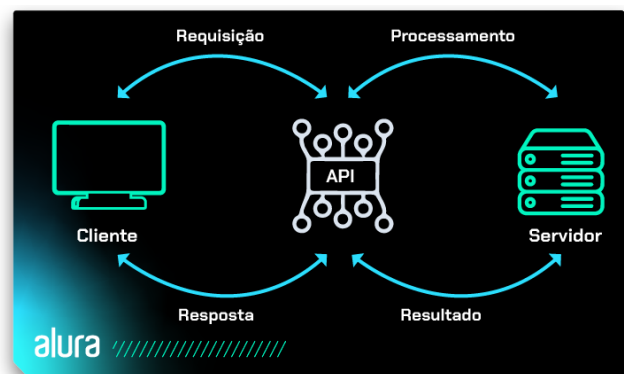
vezes com intervenção mínima ou nenhuma intervenção humana. Esses sistemas podem variar desde simples máquinas controladas por computador até redes inteiras de dispositivos interconectados, como sistemas de automação industrial, robótica, sistemas de controle de tráfego, sistemas de gerenciamento de energia, entre outros (VALENTIM *et al.*, 2012).

Os sistemas de gestão de dados desempenham um papel crucial na integração de vários tipos de informação nas organizações de saúde, garantindo uma abordagem ampla à tomada de decisões e ao atendimento ao paciente (CASTRO; LIMA; COSTA, 2020). Ao fornecer dados confiáveis e precisos, esses sistemas permitem que os profissionais em hospitais tomem decisões informadas rapidamente, levando a melhores resultados para os pacientes e eficiência operacional. Sistemas eficazes de gerenciamento de dados facilitam o fluxo contínuo de informações, levando a uma melhor coordenação de cuidados, planos de tratamento personalizados e conseqüentemente a uma melhor qualidade do atendimento ao paciente (VALENTIM *et al.*, 2012).

2.3 API (Interface de Programação de Aplicativos)

API, ou Interface de Programação de Aplicativos, é um conjunto de regras, protocolos e ferramentas que permite que diferentes *softwares* interajam entre si, incluindo especificações para solicitações e respostas de dados, formatos de mensagem, autenticação e autorização, entre outros aspectos técnicos. Conforme demonstrado na Figura 2, ela define como os componentes de *software* devem se comunicar, fornecendo uma camada de abstração que simplifica a complexidade subjacente dos sistemas (LOUZADA; CARVALHO; LARANJA, 2024).

Figura 2 - Diagrama que representa a relação de uma API com os clientes e o servidor



Fonte: Louzada, Carvalho e Laranja (2024).

No cenário atual de desenvolvimento, as APIs são essenciais para promover interoperabilidade. Ao definir padrões comuns de comunicação, elas permitem que diferentes sistemas e plataformas se integrem de maneira eficiente, independentemente de suas linguagens de programação, arquiteturas ou infraestruturas (LOUZADA; CARVALHO; LARANJA, 2024). Isso facilita a criação de ecossistemas tecnológicos mais robustos e flexíveis, nos quais os desenvolvedores podem construir sobre o trabalho de outros e combinar serviços de terceiros de forma coesa.

2.4 MVP (Produto Mínimo Viável)

O MVP, ou Produto Mínimo Viável, é uma abordagem fundamental no desenvolvimento de produtos e startups. Ele representa a versão mais simplificada de um produto que ainda é capaz de satisfazer as necessidades básicas do cliente e validar hipóteses essenciais para o sucesso do produto e é amplamente utilizado em ambientes de desenvolvimento ágil e empreendedorismo (RIES, 2012).

A essência do MVP reside na ideia de lançar um produto no mercado o mais rápido possível para testar suas hipóteses fundamentais, coletar *feedback* dos usuários e adaptar-se rapidamente às mudanças do mercado. Em vez de investir tempo e recursos significativos em desenvolver um produto completo e perfeito desde o início, o MVP permite que os empreendedores e as equipes de desenvolvimento foquem nos recursos essenciais para resolver o problema central do cliente (RIES, 2012).

É importante notar que o MVP não significa apenas lançar um produto incompleto e esperar que os usuários aceitem suas falhas. Em vez disso, trata-se de identificar os recursos essenciais que proporcionam valor imediato aos usuários e garantir que esses recursos funcionem de maneira eficaz. O MVP deve ser viável o suficiente para entregar uma experiência satisfatória, mesmo que seja simplificada em comparação com a visão final do produto.

2.5 Auto *check-in*

Sistemas de auto *check-in* (ou sistemas de autoatendimento) são sistemas automatizados projetados para permitir que os usuários realizem uma variedade de tarefas, como *check-in* em hotéis, emissão de bilhetes de transporte, compra de produtos em lojas ou realização de

transações bancárias, sem a necessidade de intervenção humana direta. Esses sistemas geralmente utilizam interfaces digitais, como telas sensíveis ao toque ou aplicativos móveis, para permitir que os usuários interajam com eles de forma simples e intuitiva, agilizando o processo e reduzindo a necessidade de atendimento pessoal (MGITECH, 2021).

Inicialmente, setores como hospedagem e aviação foram os pioneiros na implementação de sistemas de auto *check-in*. Na indústria hoteleira, aplicativos móveis e totens tornaram-se comuns, oferecendo aos hóspedes uma maneira rápida e conveniente de acessar seus quartos sem passar por recepções (VIEIRA *et al.*, 2017). A Figura 3 exibe um exemplo de totens de auto *check-in* implementados em um aeroporto, os quais vêm permitindo que passageiros realizem o *check-in* de forma autônoma, reduzindo filas e otimizando o tempo de espera (IBERIA..., 2022).

Figura 3 - Foto de totens de auto *check-in* da companhia aérea Iberia



Fonte: Antonio Heredia *apud* Iberia [...] (2022).

Com a crescente demanda por serviços de saúde e a necessidade de otimizar recursos, os sistemas de auto *check-in* têm sido cada vez mais adotados em clínicas, consultórios médicos e hospitais. Durante crises de saúde pública, como a pandemia provocada pelo COVID-19, os sistemas de auto *check-in* desempenharam um papel crucial na redução do contato físico e na minimização do risco de transmissão de doenças infecciosas, visto que a possibilidade de realizar o *check-in* remotamente antes da consulta diminui interações presenciais desnecessárias, contribuindo para a segurança tanto dos pacientes quanto dos profissionais de saúde (GRALHA *et al.*, 2022).

2.6 Tecnologias

2.6.1 JavaScript

O JavaScript é uma linguagem poderosa e versátil que permite a criação de aplicativos *web* dinâmicos e interativos. Uma de suas características distintivas é a natureza de linguagem de programação orientada a eventos e baseada em protótipos. Isso permite a criação de código assíncrono e reativo, tornando-a adequada para lidar com interações de usuário em tempo real e operações de rede. Além disso, é uma linguagem dinâmica e de tipagem fraca, o que significa que os tipos de dados são determinados em tempo de execução e as variáveis podem ser facilmente reatribuídas com diferentes tipos de valores, tornando-a altamente adaptável a uma variedade de contextos de desenvolvimento (PEREIRA; SILVÉRIO, 2023).

No desenvolvimento *web*, é essencial para a criação de interfaces de usuário interativas, e pode ser encontrada em praticamente todos os sites modernos. JavaScript é conhecida por sua sintaxe simples e flexível, e é especialmente poderosa para manipulação de elementos HTML, interatividade em páginas *web* e comunicação assíncrona com servidores. *Frameworks* como Node.js permitem que JavaScript seja executado no lado do servidor, possibilitando o desenvolvimento de aplicativos escaláveis (PEREIRA; SILVERIO, 2023).

Apesar de sua ampla popularidade e adoção, o JavaScript apresenta algumas desvantagens. A ausência de tipagem estática dificulta a manutenção e depuração de grandes projetos, aumentando a propensão a erros. Além disso, inconsistências na interpretação do código entre diferentes navegadores podem causar problemas de compatibilidade. Em termos de desempenho, o JavaScript pode ser ineficiente para operações intensivas em CPU. Historicamente, a falta de suporte nativo a módulos e a sintaxe verbosa adicionam camadas de complexidade ao desenvolvimento, tornando o gerenciamento de grandes bases de código mais desafiador (PEREIRA; SILVERIO, 2023).

2.6.2 TypeScript

TypeScript, uma linguagem desenvolvida pela Microsoft e baseada em JavaScript, tem se destacado desde 2012 por melhorar a escalabilidade e a manutenção de grandes projetos. Seus principais benefícios incluem a adição de tipos estáticos, que ajudam a detectar erros durante o desenvolvimento, e a compatibilidade total com JavaScript, facilitando a adoção em projetos existentes. A linguagem também oferece uma experiência de desenvolvimento aprimorada com ferramentas como autocompletar e refatoração, aumentando a produtividade (NEVES, 2024).

No entanto, TypeScript apresenta desafios, como uma curva de aprendizado íngreme para desenvolvedores acostumados apenas com JavaScript e a necessidade de configuração inicial, o que pode reduzir a produtividade inicialmente e representar barreiras para pequenas equipes ou projetos com prazos apertados (NASCIMENTO, 2023).

TypeScript é amplamente utilizado em grandes aplicações, como no desenvolvimento do Angular, e tem ganhado popularidade no desenvolvimento *back-end* com Node.js, proporcionando benefícios de segurança e manutenibilidade. A comunidade e o ecossistema de TypeScript estão crescendo rapidamente, e a Microsoft continua a lançar atualizações regulares que introduzem novas funcionalidades e melhorias (NEVES, 2024).

2.6.3 React.js

ReactJS, uma biblioteca JavaScript desenvolvida pelo Facebook em 2013, é amplamente utilizada para a construção de interfaces de usuário dinâmicas e interativas. Seus principais benefícios incluem a utilização do Virtual DOM, que melhora o desempenho da aplicação ao minimizar operações no DOM real, e a arquitetura baseada em componentes, que promove a reutilização de código e a manutenção de projetos complexos (NEVES, 2023).

O ReactJS também apresenta uma curva de aprendizado relativamente suave para desenvolvedores familiarizados com JavaScript, devido à sua sintaxe declarativa e à simplicidade de sua API. A comunidade ativa e o rico ecossistema de bibliotecas e ferramentas, como React Router e Redux, facilitam o desenvolvimento de aplicações robustas e escaláveis (NEVES, 2023).

Por outro lado, esta biblioteca requer a configuração de um ambiente de desenvolvimento adequado, o que pode ser complicado para iniciantes. Além disso, a rápida evolução da biblioteca e das ferramentas associadas exige que os desenvolvedores se mantenham constantemente atualizados (FREIRE, 2019).

ReactJS é amplamente adotado em grandes aplicações *web* devido à sua eficiência e flexibilidade. Empresas como Facebook, Instagram e Netflix utilizam ReactJS em seus produtos, demonstrando sua capacidade de suportar aplicações de alta demanda. O suporte contínuo da comunidade e as melhorias frequentes garantem que ReactJS permaneça relevante e adaptável às necessidades modernas de desenvolvimento (NEVES, 2023).

2.6.4 Node.js

A tecnologia Node.js é uma plataforma de código aberto construída sobre o motor JavaScript V8 do Google Chrome. Sua arquitetura é orientada a eventos e não bloqueante, ou seja, permite que os programas continuem executando outras tarefas enquanto aguardam a conclusão de operação de E/S (Entrada/Saída). Essa abordagem permite que o Node.js lide de forma eficiente com grandes volumes de solicitações de entrada e saída, como acesso a arquivos ou consultas a bancos de dados (BESSA, 2023).

Sua eficiência no manuseio de operações de E/S não bloqueantes tornou-o ideal para aplicações *web* em tempo real, como chats, jogos online e colaboração simultânea. Além disso, o Node.js tem sido amplamente utilizado em microsserviços, APIs, aplicações de streaming de dados, e muito mais. O ecossistema do Node.js é alimentado pelo npm, o maior repositório de *software* do mundo, que hospeda milhões de pacotes prontos para uso. Isso não apenas acelerou o desenvolvimento de *software*, mas também promoveu uma cultura de compartilhamento e colaboração entre desenvolvedores (BESSA, 2023).

Embora o Node.js ofereça muitos benefícios, o modelo de thread único de Node.js pode ser uma limitação em aplicações que exigem processamento intensivo de CPU, pois pode causar bloqueios e afetar o desempenho. A natureza assíncrona e a utilização extensiva de *callbacks* podem levar ao chamado "*callback hell*", tornando o código difícil de ler e manter. Além disso, devido à sua natureza relativamente jovem em comparação com outras tecnologias de servidor, pode haver uma curva de aprendizado íngreme para desenvolvedores acostumados a ambientes mais tradicionais, como Java ou .NET (LIMA, 2023).

2.6.5 MySQL

MySQL é um sistema de gerenciamento de banco de dados relacional de código aberto profundamente disseminado no mercado de tecnologias, cuja estrutura típica consiste em um servidor MySQL, onde reside o núcleo do *software* de banco de dados, e os clientes que se conectam a ele para acessar e manipular os dados (ORACLE, 2024).

Esta tecnologia oferece uma ampla gama de recursos que o tornam uma escolha popular entre desenvolvedores e administradores de banco de dados, como o suporte para transações ACID (Atomicidade, Consistência, Isolamento e Durabilidade), replicação para alta disponibilidade e escalabilidade, segurança avançada com autenticação e criptografia, suporte

para várias linguagens de programação, como PHP, Python e Java, e uma comunidade ativa que fornece suporte e desenvolvimento contínuo (VYAS, 2023).

O MySQL é amplamente utilizado em uma variedade de cenários e setores. Ele alimenta muitos dos sites mais populares da internet, incluindo WordPress, Facebook, Twitter e YouTube, onde é usado para armazenar e gerenciar grandes volumes de dados de forma eficiente. Além disso, o MySQL é frequentemente empregado em sistemas de comércio eletrônico, aplicativos corporativos e sistemas de gerenciamento de conteúdo (ORACLE, 2024).

Entretanto, problemas de desempenho e escalabilidade podem surgir em aplicações com alto volume de operações simultâneas de leitura e escrita. A falta de suporte completo a funcionalidades avançadas de SQL, o suporte parcial a JSON e o seu sistema de bloqueio menos sofisticado podem ser limitantes. Além disso, alguns recursos avançados e otimizações estão disponíveis apenas nas versões pagas, limitando as opções para projetos com orçamento reduzido (MARQUES, 2023).

3 METODOLOGIA

Nesta seção são descritas as etapas pertinentes à apresentação e implementação dos requisitos funcionais e técnicos do protótipo. Além disso, a elaboração é descrita a partir das seguintes etapas: design do *layout*, construção do banco de dados, construção do servidor *back-end*, construção do cliente *front-end* e implementações de funcionalidades específicas, como a integração com o Google Drive, o *download* de PDFs, a construção de um processo de automação no *Front-end*, publicação do sistema para acesso remoto e validação do sistema a partir de um formulário.

3.1 Apresentação dos requisitos

Os requisitos apresentados a seguir foram divididos em duas frentes: requisitos funcionais e requisitos técnicos. Os requisitos funcionais são as regras de negócio que o sistema precisa atender para garantir seu funcionamento, enquanto os requisitos técnicos do sistema constituem nas ferramentas necessárias para a confecção do mesmo e para o atendimento das regras de negócio estabelecidas.

3.1.1 Requisitos funcionais

O sistema deve ser uma plataforma de comunicação direta entre paciente, enfermeiro e médico, sendo os requisitos funcionais do mesmo definidos por meio do levantamento das necessidades encontradas em hospitais da rede pública e privada de Sorocaba feito a partir de entrevistas com profissionais da saúde. Tais requisitos são listados a seguir:

- Sinalizar a presença do paciente no hospital;
- Anexar e acessar exames do paciente;
- Registrar o prontuário do paciente;
- Registrar a prescrição médica;
- Ter uma interface de controle de usuários.

Diante dos requisitos gerais apresentados, houve a necessidade de destrinchá-los em fluxos, isto é, ações que o usuário pode realizar na plataforma, com o intuito de definir os requisitos funcionais específicos.

O sistema possui 4 fluxos para o seu funcionamento baseados em seus níveis de usuário, são eles:

- Administrador;
- Médico;
- Enfermeiro;
- Paciente.

Tais níveis são explicados detalhadamente no decorrer dos tópicos seguintes.

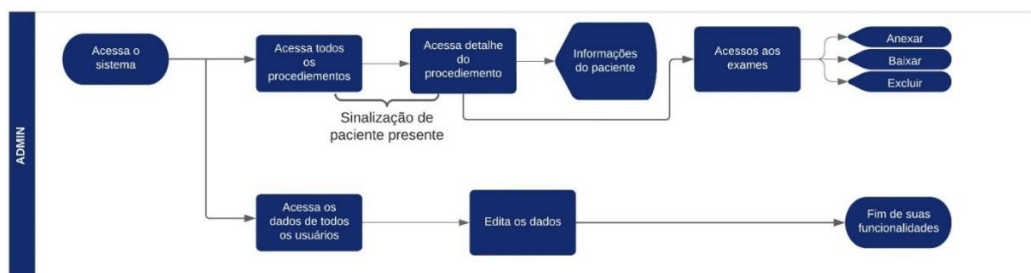
3.1.1.1 Requisitos do fluxo Administrador

O primeiro nível a ser tratado é o nível de Administrador, o qual corresponde à parte administrativa do hospital ou clínica e tem o intuito de acessar as informações de todos os envolvidos na realização do procedimento. Os requisitos deste nível são:

- Acessar todos os registros dos pacientes, enfermeiros e médicos, sem possibilidade de alteração dos laudos;
- Alterar todos os dados cadastrais de todos os usuários.

A partir dos requisitos definidos, idealizou-se o fluxograma para o usuário do tipo Administrador, apresentado na Figura 4.

Figura 4 - Fluxo idealizado para o usuário do tipo Admin



Fonte: Autoria própria.

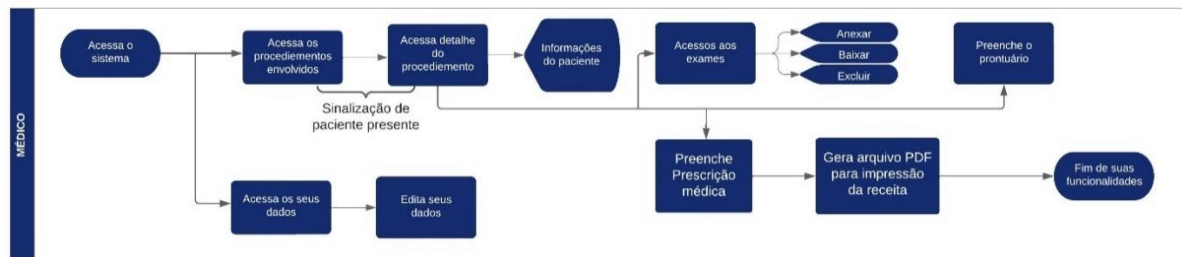
3.1.1.2 Requisitos do fluxo Médico

O segundo nível é o de Médico que, como o próprio nome indica, permite o acesso do médico, onde ele tem acesso aos procedimentos em que está envolvido e pode ver se o paciente está presente no recinto. Os requisitos deste nível são:

- Sinalizar que o paciente está no recinto;
- Preencher prontuário eletrônico;
- Preencher, alterar e fazer *download* prescrição médica;
- Acessar, registrar e excluir exames do paciente;
- Editar os próprios dados do médico.

A partir dos requisitos definidos, idealizou-se o fluxograma para o usuário do tipo Médico, apresentado na Figura 5.

Figura 5 - Fluxo idealizado para o usuário do tipo Médico



Fonte: Autoria própria.

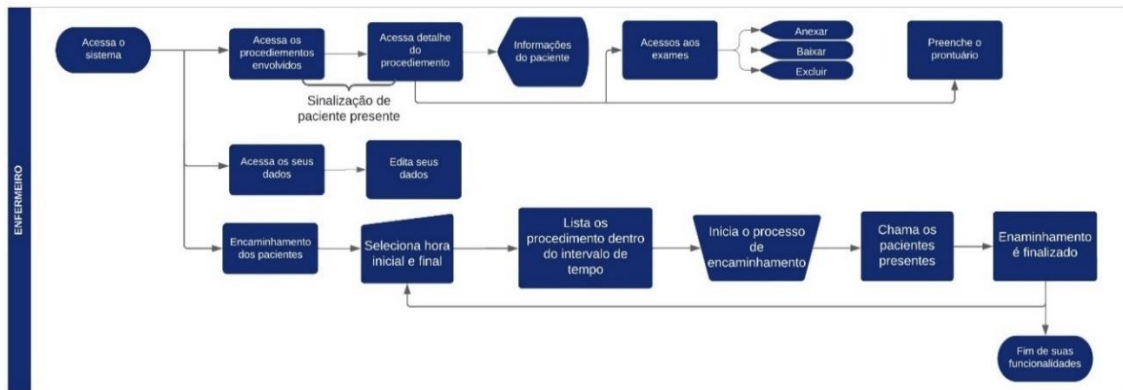
3.1.1.3 Requisitos do fluxo Enfermeiro

O terceiro nível é o de Enfermeiro que, como o próprio nome indica, permite o acesso do enfermeiro, tem praticamente a mesma finalidade que o segundo nível, porém tem a opção de encaminhar aqueles pacientes que estão presentes. Os requisitos deste nível são:

- Sinalizar que o paciente está no recinto;
- Preencher prontuário eletrônico;
- Acessar, registrar e excluir exames do paciente;
- Filtrar os procedimentos em que está envolvido, para um determinado intervalo de tempo (horas), fazendo que haja um encaminhamento automático dos pacientes presentes;
- Editar os próprios dados do enfermeiro.

A partir dos requisitos definidos, idealizou-se o fluxograma para o usuário do tipo Enfermeiro, apresentado na Figura 6.

Figura 6 - Fluxo idealizado para o usuário do tipo Enfermeiro



Fonte: Autoria própria.

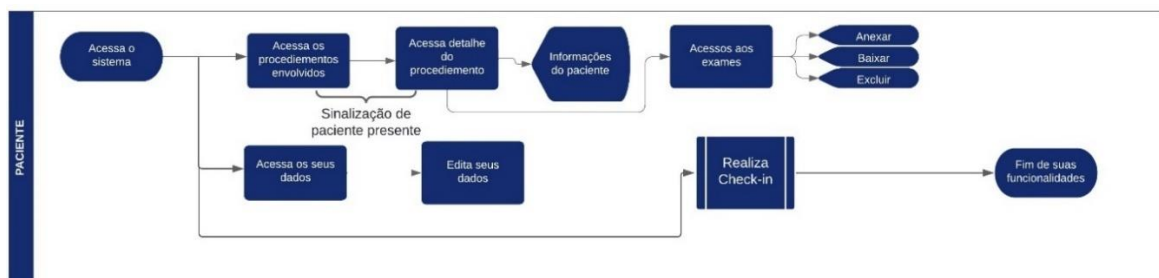
3.1.1.4 Requisitos do fluxo Paciente

O quarto e último nível é o de Paciente que, como o próprio nome indica, permite o acesso do paciente, permitindo-o acompanhar os procedimentos no qual será submetido, agilizar seu atendimento e esclarecer eventuais dúvidas quanto ao seu procedimento. Os requisitos deste nível são:

- Sinalizar para o hospital que está no recinto;
- Acessar, registrar e excluir seus exames;
- Acessar e consultar seus procedimentos e acessar a prescrição médica;
- Editar seus próprios dados;

A partir dos requisitos definidos, idealizou-se o fluxograma para o usuário do tipo Paciente, apresentado na Figura 7.

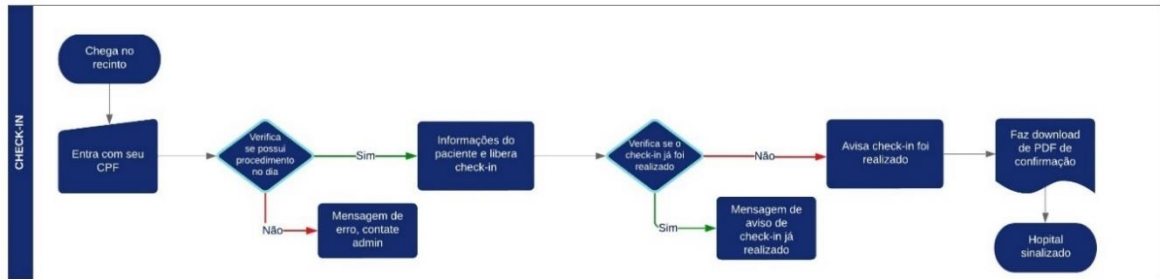
Figura 7 - Fluxo idealizado para o usuário do tipo Paciente



Fonte: Autoria própria.

E para o processo de *Check-in* presente no fluxo do Paciente, obteve-se o fluxograma idealizado para tal função, apresentado na Figura 8.

Figura 8 - Fluxo idealizado para o processo de *Check-in*



Fonte: Autoria própria.

3.1.2 Requisitos técnicos

Com a definição dos requisitos funcionais, foi necessário determinar os requisitos técnicos, isto é, quais são as ferramentas que atenderão os fluxos e as regras de negócios impostas anteriormente. Os seguintes requisitos técnicos foram definidos de maneira geral, como segue:

- O desenvolvimento da interface e telas de usuário deve ser feito com React JS;
- O banco de dados deve ser um relacional, sendo escolhido o MySQL;
- O desenvolvimento de funcionalidades do servidor e comunicação com o banco de dados deve ser feito com o NodeJS;
- O sistema deve ser capaz de gerar arquivo em PDF, para a identificação do paciente e gerar receita médica;
- O sistema deve armazenar os exames do paciente via Google Drive;
- O sistema deve conter ao menos uma automação para o processo de encaminhamento de pacientes presentes;
- O sistema deve ser publicado para que possa ser acessado em outros locais.

Diante destes requisitos, foi possível traçar uma ordem de desenvolvimento, já que para a confecção de um sistema *web* há necessidade de um preparo para a obtenção de uma espécie de rascunho. A sequência a seguir visa detalhar este processo de implementação dos requisitos técnicos.

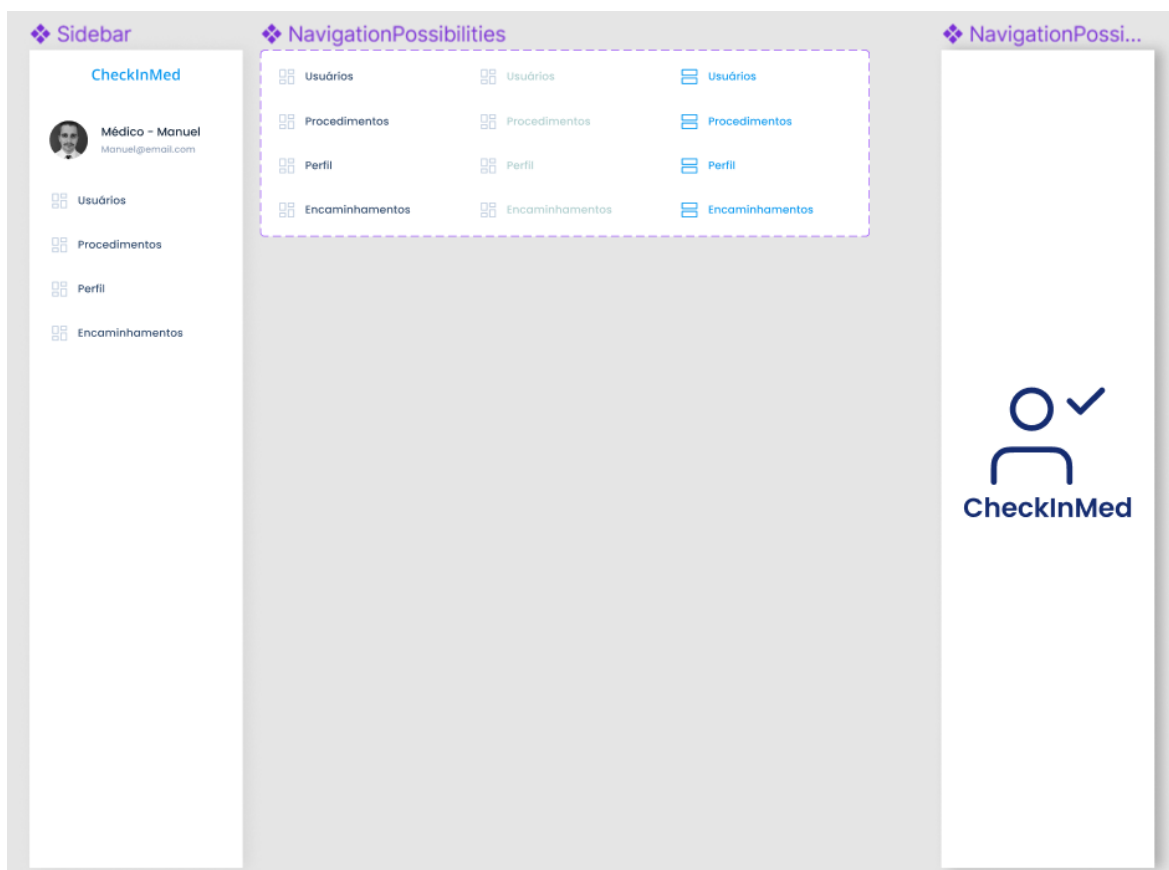
3.1.2.1 Construção das telas do sistema CheckInMed

O sistema foi denominado CheckInMed e projetado utilizando-se o Figma, uma plataforma que promove eficiência e facilidade na prototipagem de *layouts* e design de telas de aplicativos mobile ou *web* (VILLAIN; SILVEIRA, 2023).

O design das telas do protótipo foi feito a partir de um modelo de projeto disponível na área de exploração do Figma, desenvolvido a partir de um modelo que se assemelhasse a um sistema de saúde moderno (FIGMA, 2024).

Deu-se continuidade à construção das telas a partir da componentização, processo que permite o reaproveitamento de elementos visuais em diferentes telas, como por exemplo os menus laterais, que foram utilizados em todas as telas do projeto a fim de facilitar a navegação entre elas, conforme mostra a Figura 9.

Figura 9 - Componentes visuais dos menus laterais construídos no Figma, sendo que o menu à direita é o que foi utilizado nas páginas Inicial e *Check-in* e o menu à esquerda é o que foi utilizado nas demais páginas



Fonte: Autoria própria.

As telas construídas no Figma são listadas no Quadro 1 e podem ser visualizadas no apêndice A.

Quadro 1 - Definição e descrição das telas construídas no Figma

Tela	Descrição
Inicial ou <i>Home</i>	Direciona os usuários para a área de <i>Check-in</i> , proporcionando uma experiência intuitiva de navegação
<i>Check-in</i>	Detalha cada etapa do processo, garantindo que o usuário compreenda claramente as ações a serem executadas
Listagem de Usuários	Exibe uma tabela com informações essenciais dos usuários e permite o redirecionamento para a tela de Detalhe de Usuários
Listagem de Procedimentos	Exibe uma tabela com informações essenciais dos procedimentos e permite o redirecionamento para a tela de Detalhe de Procedimento
Detalhe de Usuário	Detalha todas as informações pertinentes ao usuário em componentes expansíveis e permite a edição dos mesmos
Detalhe de Procedimento	Detalha todas as informações pertinentes ao procedimento em componentes expansíveis e permite a edição de alguns itens
Encaminhamento	Exibe elementos da lista de procedimentos e demonstra como os elementos seriam configurados para executar a automação

Fonte: Autoria própria.

Com os modelos de telas definidas no Figma, foi possível preparar o ambiente da aplicação *front-end*, com o intuito de desenvolver interfaces interativas para o sistema em questão.

O *front-end*, ou "parte da frente", é a parte de um sistema de *software* com a qual os usuários interagem diretamente e que envolve o design, a interface do usuário e a experiência do usuário. Em termos técnicos, o *front-end* é construído usando linguagens de marcação (como HTML), estilização (como CSS) e programação (como JavaScript) para criar a interface com a qual os usuários interagem em seus navegadores ou dispositivos. Em resumo, é tudo o que os usuários veem e com o que interagem em um aplicativo ou site (SOUTO, 2024).

Para a configuração da aplicação *front-end*, optou-se por utilizar ReactJS juntamente com TypeScript. Além disso, organizou-se tal aplicação em uma estrutura de pastas e construiu-se o objeto de rotas para disponibilizar os componentes das telas da aplicação, isto é, funções que recebem propriedades como entrada e retornam elementos que descrevem a interface do usuário.

As rotas no *front-end* se referem ao roteamento das páginas dentro da aplicação *web*, isto é, a associação dos caminhos URL para a renderização dos componentes visuais que compõem uma determinada página. As rotas definidas são listadas no Quadro 2.

Quadro 2 - Definição das rotas construídas na aplicação *front-end*

Rota	Descrição
"/"	Esta rota compõe a página inicial (<i>Home</i>).
"/checkin"	Esta rota compõe a página de <i>Check-in</i> .
"/users"	Esta rota compõe a página de listagem de Usuários.
"/users/:id"	Esta rota compõe a página de detalhes de Usuário.
"/medicalProcedures"	Esta rota compõe a página de listagem de Procedimentos.
"/medicalProcedures/:id"	Esta rota compõe a página de detalhes de Procedimento.
"/referral"	Esta rota compõe a página de Encaminhamento.

Fonte: Autoria própria.

3.1.2.2 Modelagem do banco de dados relacional

O banco de dados escolhido para o projeto foi o MySQL, uma opção amplamente reconhecida e utilizada na comunidade de desenvolvimento de *software* devido a sua confiabilidade e eficiência como banco de dados relacional (HOSTMÍDIA, 2024).

Para modelar a estrutura do banco de dados, optou-se por utilizar o *software* dbdiagram.io, plataforma gratuita que oferece uma representação visual dos relacionamentos entre as tabelas e disposição dos dados e seus respectivos tipos, servindo também como uma documentação técnica do projeto. Parte da modelagem construída é apresentada na Figura 10.

Figura 10 - Parte de Usuários presente no modelo do banco de dados construído no dbdiagram.io



Fonte: Autoria própria.

O restante do modelo do banco de dados construído no dbdiagram.io pode ser encontrado no apêndice B.

3.1.2.3 Desenvolvimento das funcionalidades do servidor e comunicação com o banco de dados

O *back-end*, ou "parte de trás", de um sistema de *software* é a parte que lida com o processamento dos dados e a lógica de negócios. É responsável pelo armazenamento, recuperação e manipulação de dados, geralmente em um servidor remoto. Isso envolve o desenvolvimento de servidores, bancos de dados, aplicativos e APIs que permitem que o *front-end* (a parte com a qual os usuários interagem) se comunique com o sistema e receba os dados necessários para exibição. Em suma, o *back-end* é a "engrenagem" invisível que torna as aplicações *web* e sistemas funcionais (EWALLY, 2023).

A aplicação *back-end* foi configurada utilizando Node.js com TypeScript, oferecendo robustez e tipagem estática para o código, e organizada em uma estrutura de pastas que abrange os recursos essenciais para sua execução.

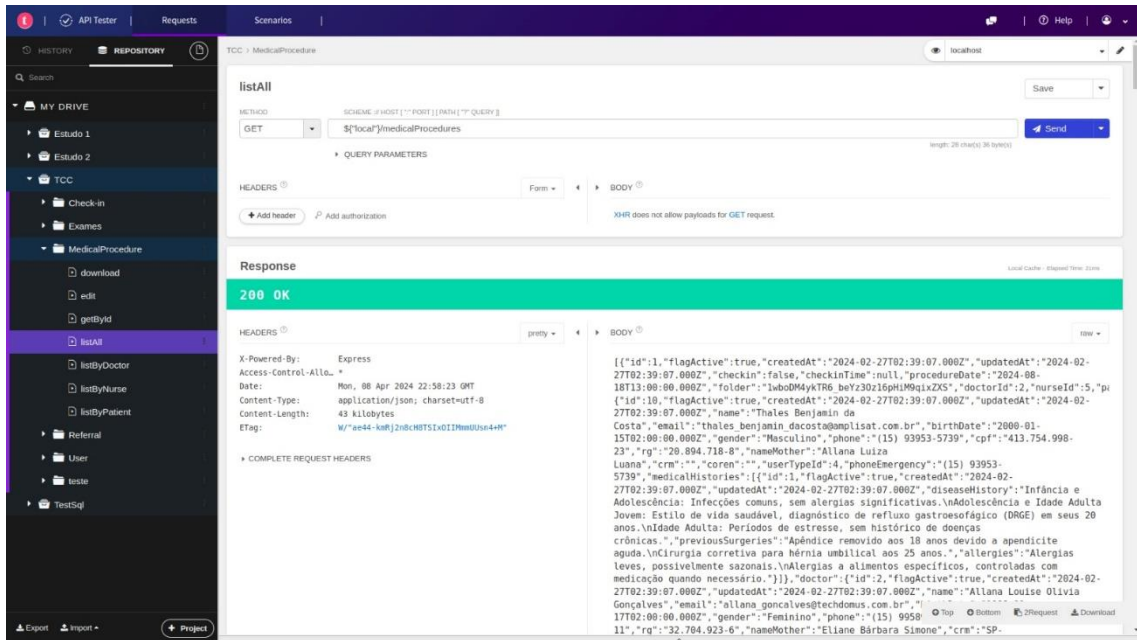
Para gerenciar o banco de dados, optou-se pelo TypeORM, um tipo de ferramenta ORM (*Object-Relational Mapping* ou Mapeador de Objeto Relacional) que simplifica a interação com o banco de dados por meio de código. Utilizando-se tal ferramenta, configurou-se a comunicação com o banco de dados (tal comunicação foi feita utilizando o protocolo TCP/IP a partir do *localhost* e da porta 3306) de forma a garantir a integração entre a aplicação e os dados armazenados, os quais são interpretados a partir do método de modelos, desenvolvido para refletir as tabelas do banco de dados em forma de código. Além disso, foram configurados os relacionamentos entre as tabelas, garantindo uma representação precisa da estrutura de dados.

A utilização de um ORM pode causar sobrecarga no sistema, especialmente em cenários com um grande fluxo de usuários, como 10000 usuários por mês. No entanto, para uma quantidade pequena de pessoas, pensadas inicialmente para 50 usuários no sistema, a sobrecarga não afetará significativamente o desempenho do sistema. Alternativas para mitigar essa sobrecarga incluem o uso de técnicas de otimização de consultas, cache de dados, implementação de índices adequados no banco de dados e até mesmo a consideração de ORM mais leves e mais eficientes para casos de uso específicos. Utilizar consultas SQL diretas em casos em que o desempenho é crítico também é uma opção viável. Essas abordagens podem ajudar a minimizar a sobrecarga do sistema e garantir um desempenho mais consistente, mesmo em ambientes com um grande volume de dados (MATEUS, 2023).

Para disponibilizar funcionalidades para os usuários da aplicação *front-end*, foram desenvolvidas rotas, ou *endpoints* (os quais consistem em pontos de comunicação em sistemas de *software* ou redes que permitem a troca de dados e solicitações entre diferentes partes de um sistema), que são consumidas por meio de requisições HTTP, que transportam dados no formato JSON. Este procedimento se configura como uma API do sistema.

Para garantir que as funcionalidades criadas estivessem de acordo com suas funções e com o devido retorno de dados para serem consumidas no *front-end*, utilizou-se a plataforma Talend API Tester, um *software* presente nas extensões do navegador do Google Chrome. Reforça-se que tal *software* não é ideal para testes de carga de dados e sim para garantir o funcionamento dos *endpoints* e seu devido retorno. Sua utilização permitiu a realização dos testes das rotas, proporcionando serviços de HTTP e retornando as respostas correspondentes a cada requisição ao servidor. A Figura 11 mostra a aparência da plataforma.

Figura 11 - *Endpoint* de listagem de procedimento médicos e seu retorno no formato JSON feito no *software* Talend API Tester, utilizado para testar os *endpoints* disponibilizados pelo servidor



Fonte: Autoria própria.

Para atender ao requisito de armazenamento de exames do paciente via Google Drive, realizou-se a integração da funcionalidade de armazenamento na aplicação *back-end* seguindo as etapas listadas a seguir:

- Configurou-se o Google Cloud Console, plataforma para habilitação dos serviços de desenvolvimento da Google, para habilitar o serviço da API do Google Drive, permitindo o acesso e utilização dos recursos oferecidos por essa API na aplicação.
- Criou-se um objeto JSON de autenticação do Google utilizando credenciais do tipo *service account* para conceder permissões de acesso à API através de códigos externos.
- Com os itens de configuração concluídos, foram implementadas as funcionalidades de *upload*, listagem, *download* e exclusão de arquivos do Google Drive diretamente na aplicação *back-end*.
- Os arquivos foram organizados por pastas no Google Drive, utilizando IDs de pastas únicas para associar cada procedimento a uma pasta específica. Essas pastas foram compartilhadas com o e-mail gerado por uma credencial *service account*, garantindo acesso aos arquivos para cada procedimento.

- Para efetuar o *download* de arquivos, foram utilizadas IDs das pastas e dos arquivos correspondentes no Google Drive, permitindo a recuperação e disponibilização dos arquivos desejados com base em suas localizações no armazenamento do Google Drive.

Outro requisito também foi implementado na aplicação *back-end*: a geração de arquivos em PDF. Tal funcionalidade foi desenvolvida por meio da biblioteca *puppeteer*, que permite a conversão dos *templates* HTML em arquivos PDF. Para a apresentação do arquivo PDF gerado, duas maneiras foram implementadas: *download* do arquivo na máquina do usuário e carregamento direto na página do navegador, retornando o arquivo em tempo real no corpo da resposta.

3.1.2.4 Automação do processo de encaminhamento de pacientes

Realizou-se a automação do processo de *check-in* na aplicação *front-end* de forma a atender ao requisito de encaminhamento automatizado de pacientes presentes e de ilustrar um recurso adicional que pode ser implementado no processo de auto *check-in* para agregar valor ao protótipo.

Para o funcionamento da automação são requeridas uma hora inicial e uma hora final, representando o intervalo de tempo durante o qual a mesma será executada. A diferença entre esses dois horários é calculada em milissegundos e foi chamada de *timeout*.

Após o cálculo do *timeout*, a função *setInterval*, nativa do JavaScript, é chamada. Essa função executa repetidamente outra função a cada intervalo de tempo especificado. No caso, a função executada é responsável por chamar uma rota no *back-end* para consultar e retornar os procedimentos que possuem *check-in* verdadeiro dentro do intervalo de horas determinado.

O tempo especificado para a execução do *setInterval* é de 1 minuto. Essa escolha foi feita para garantir tempo suficiente para chamar a rota e respeitar o intervalo de tempo especificado. Isso é importante, pois pode haver casos em que a função demore a ser executada devido a diversos fatores, como a carga do servidor ou a complexidade das consultas realizadas pela rota.

Após a execução do *setInterval*, é salvo um número que representa o identificador do intervalo na memória. Esse identificador pode ser entendido como um ID do *setInterval* e é usado para controlar e gerenciar a execução do intervalo de tempo.

Em seguida, é executada a função *setTimeout*, também nativa do JavaScript. Essa função é responsável por executar uma determinada função uma única vez quando o tempo

especificado termina. Ela requer como parâmetro o tempo em milissegundos, que é o *timeout* calculado no início do processo.

O objetivo do *setTimeout* é encerrar automaticamente o processo de automação após o tempo especificado, garantindo que o intervalo de execução definido pelo *setInterval* seja respeitado. Após a execução do *setTimeout*, o identificador associado a ele também é salvo na memória.

Após as funções que executam a automação começarem a funcionar, algumas informações são salvas no *sessionStorage*. O *sessionStorage* é uma forma de armazenamento de dados de forma persistente durante o uso do próprio navegador. Esses dados não são sensíveis e incluem os dados listados no Quadro 3.

Quadro 3 - Definição das variáveis que compõem a automação presente na página de encaminhamento

Variável	Descrição
intervalId	O ID do processo inicializado no <i>setInterval</i> .
timeoutId	O ID do processo inicializado no <i>setTimeout</i> .
job	Indica se a automação está ativada ou desativada.
selectedTimeStart	A hora inicial escolhida pelo usuário.
selectedTimeEnd	A hora final escolhida pelo usuário.
startDate	O valor do objeto <i>Date</i> da hora inicial para ser enviado ao servidor.
endDate	O valor do objeto <i>Date</i> da hora final para ser enviado ao servidor.

Fonte: Autoria própria.

3.1.2.5 Publicação das aplicações

Para realizar o *deploy* do protótipo (processo de disponibilizar ou publicar uma aplicação ou *software* em um ambiente de produção, tornando-o acessível aos usuários finais), optou-se por hospedá-lo nas seguintes plataformas de nuvem: Render e Vercel. O Render foi escolhido para hospedar o banco de dados online e a aplicação *back-end*, enquanto a Vercel foi utilizada para hospedar a aplicação *front-end*. Essa escolha foi feita devido ao suporte oferecido por essas plataformas às tecnologias utilizadas no desenvolvimento das aplicações.

A publicação das aplicações foi feita em duas etapas. Primeiramente, o banco de dados e a aplicação *back-end* foram publicados. Isso permitiu obter o caminho no qual o servidor estaria rodando, possibilitando a comunicação com o *front-end*. Em seguida, a aplicação *front-end* foi publicada, garantindo assim o correto funcionamento de todo o sistema após o *deploy* completo.

3.1.2.6 Validação do protótipo

A validação do protótipo foi conduzida da seguinte maneira: quinze pessoas experimentaram o sistema, incluindo médicos, enfermeiros, administradores de hospitais e profissionais de outras áreas, configurando-os como pacientes. Os candidatos selecionados receberam acesso ao sistema, publicado por meio de *deploys* das aplicações *front-end*, *back-end* e banco de dados, juntamente com uma lista de CPFs fictícios (sendo 1 registro para o Admin, 3 registros de Médicos, 5 registros de Enfermeiros e 10 Pacientes) previamente registrados no banco de dados. Esses CPFs foram essenciais para autenticar o acesso ao sistema e permitir a visualização do fluxo de usuário correspondente.

A utilização de nomes, CPFs, e demais dados pessoais e médicos fictícios foi feita a fim de garantir o respeito à LGPD (Lei Geral de Proteção de Dados) e a não utilização de dados reais sensíveis, como, por exemplo, informações médicas sigilosas.

Antes de iniciar os testes, os candidatos foram instruídos a respeito do objetivo e das funcionalidades presentes no sistema. Eles foram orientados a priorizar a exploração do fluxo de usuário correspondente a sua área de atuação, mas também tiveram a liberdade de experimentar outros fluxos, caso desejassem. Além disso, durante os testes, os participantes foram acompanhados para o esclarecimento de eventuais dúvidas pertinentes ao sistema.

Ao finalizar os testes, os candidatos preencheram um formulário destinado à coleta de suas considerações sobre o sistema. Este formulário abordou aspectos como usabilidade e funcionalidade para os diferentes tipos de usuário. Ademais, os participantes avaliaram a inovação, eficiência, impacto, satisfação e aceitação do sistema, podendo também pontuar problemas encontrados, sugerir melhorias e justificar as opiniões fornecidas.

4 RESULTADOS E DISCUSSÕES

Nesta seção são apresentados e discutidos os resultados pertinentes à construção e população do banco de dados, implementação dos *endpoints* fornecidos pela API construída no *back-end*, desenvolvimento das telas no *front-end* com base no modelo definido no Figma para utilização dos usuários, automação do processo de encaminhamento de pacientes e validação do sistema projetado.

4.1 Construção e população do banco de dados

Foi possível gerar *script* para a criação do banco e de tabelas a partir da plataforma dbdiagram.io, ao qual oferece suporte MySQL. Além do *script* de criação fornecido pela plataforma de modelagem do banco de dados, foram criados *scripts* manualmente para popular o banco de dados e, assim, ter uma carga de dados para o desenvolvimento de exemplos e testes de verificação do tratamento de dados. A geração de dados priorizou a ampliação da quantidade de usuários registrados no sistema, tornando os testes mais verossímeis.

Durante esse processo, houve uma equalização dos dados, garantindo que para cada nível de acesso (médico ou enfermeiro), a quantidade de procedimentos médicos fosse a mesma. Para o nível de acesso dos pacientes, não se estabeleceu um número fixo de procedimentos, permitindo que alguns pacientes fossem associados a mais de um procedimento. Os *scripts* utilizados podem ser consultados nos apêndices C e D.

4.2 API fornecida pelo *back-end* do sistema

A API constitui-se de um conjunto de *endpoints* ou rotas e configura-se como resultado da aplicação *back-end*, na qual foram implementados os recursos de Usuários, Procedimentos, *Check-in*, Encaminhamento e Exames. Estas rotas são descritas a seguir.

Antes de citar os *endpoints* construídos, vale ressaltar que ao utilizar uma rota GET com parâmetros como ID pode-se ter problemas de segurança, por isso, é essencial implementar validação e sanitização de entradas, autenticação e autorização de usuários, uso de consultas preparadas para evitar injeção de SQL, limitação de dados sensíveis retornados, proteção contra-ataques de força bruta com *rate limiting*, e monitoramento de logs. Alternativamente, utilizar uma rota POST para transmitir dados sensíveis no corpo da requisição, combinado com

autenticação e autorização rigorosas, pode oferecer uma camada adicional de segurança em comparação com o método GET. Porém para os fins abordados do sistema como protótipo em apenas demonstrar o funcionamento das funcionalidades, o método GET com parâmetro nos *endpoints* não foi necessário tal tratamento de segurança (ANDRADE, 2019).

A rota de Usuários, identificada por "/users", gerencia as operações relacionadas aos usuários do sistema e possui funcionalidades como: listar todos os usuários, consultar detalhes de um usuário específico, editar informações de um usuário por meio de seu ID único e consultar CPF para confirmação de usuário e tipo. O detalhamento das rotas de Usuários mencionadas é feito no Quadro 4.

Quadro 4 - Relação das rotas de Usuários disponibilizadas pelo servidor

Rotas de Usuários		
<i>Endpoint</i>	Método HTTP	Função
/users	GET	Lista todos os usuários registrados, trazendo seus dados
/users/:id	GET	Ao receber o 'id' fornecido, os dados do usuário que possui essa identificação são retornados
/users/:id	PUT	Atualiza os dados do usuário que possui o 'id' passado na requisição
/users/validateUser	POST	Recebe no corpo da requisição um CPF para validá-lo

Fonte: Autoria própria.

A rota de Procedimentos, identificada por "/medicalProcedures", gerencia as operações relacionadas aos procedimentos médicos. Essa rota permite listar todos os procedimentos, consultar detalhes de um procedimento específico e editar informações de um procedimento, como prontuários e prescrições médicas. O detalhamento das rotas mencionadas é feito no Quadro 5.

Para a geração de receituários médicos na página do navegador, desenvolveu-se um *template* HTML que integra informações relevantes, como dados do paciente, identificação do médico e a lista de medicamentos prescritos. Além disso, foi implementada a rota /medicalProcedures/download/:id no sistema, a qual é acionada com o fornecimento do ID do procedimento como entrada e utiliza os dados associados a ele para preencher o *template* HTML. Como resultado desse processo, um arquivo PDF é gerado e enviado como resposta à requisição HTTP feita pelo usuário, sendo apresentada em uma página no navegador, sem a

necessidade de realizar diretamente o *download* do arquivo. Tal arquivo da receita médica pode ser encontrado no apêndice E.

Quadro 5 - Relação das rotas de Procedimentos Médicos disponibilizadas pelo servidor

Rotas de Procedimento Médico		
<i>Endpoint</i>	Método HTTP	Função
/medicalProcedures	GET	Lista todos os procedimentos médicos registrados, trazendo seus dados
/medicalProcedures/listByDoctor/:doctorId	GET	Lista todos os procedimentos médicos registrados relacionados ao médico que possui o 'id' de usuário informado pelo parâmetro 'doctorId'
/medicalProcedures/listByNurse/:nurseId	GET	Lista todos os procedimentos médicos registrados relacionados ao enfermeiro que possui o 'id' de usuário informado pelo parâmetro 'nurseId'
/medicalProcedures/listByPatient/:patientId	GET	Lista todos os procedimentos médicos registrados relacionados ao paciente que possui o 'id' de usuário informado pelo parâmetro 'patientId'
/medicalProcedures/:id	GET	Ao receber o 'id' fornecido, os dados do procedimento médico que possui essa identificação são retornados
/medicalProcedures/:id	PUT	Atualiza os dados do procedimento médico que possui o 'id' passado na requisição
/medicalProcedures/download/:id	GET	Recebe o 'id' do procedimento médico e gera o PDF da prescrição médica

Fonte: Autoria própria.

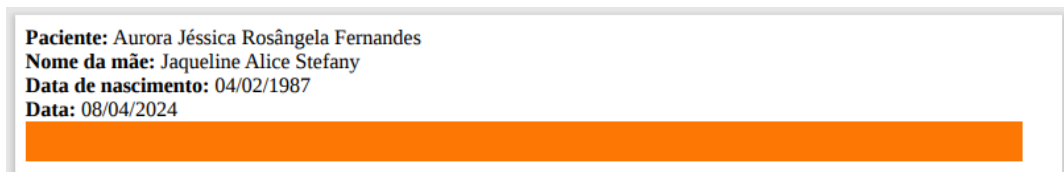
A rota de *Check-in*, identificada por "/checkin", trata do registro de presença de um paciente em um procedimento marcado para o dia atual. Suas funcionalidades incluem verificar se o paciente possui procedimento marcado para o dia, atualizar o dado (tipo de identificação booleana) de *check-in* nos procedimentos e disponibilizar o *download* de uma guia de identificação para o paciente. O detalhamento das rotas mencionadas é feito no Quadro 6.

Quadro 6 - Relação das rotas de *Check-in* disponibilizadas pelo servidor

Rotas de <i>Check-in</i>		
<i>Endpoint</i>	Método HTTP	Função
/checkin/search	POST	Recebe um CPF no corpo da sua requisição e retorna as informações de <i>check-in</i> para procedimentos médicos do dia
/checkin/checkin	PUT	Atualiza a o <i>check-in</i> do procedimento médico e gera PDF de confirmação

Fonte: Autoria própria.

Para a geração de arquivo PDF a partir do método de *download* (rota /checkin/checkin), desenvolveu-se um *template* base com algumas informações do paciente e do procedimento. Assim que confirmado, o arquivo PDF é baixado para o usuário, simulando a confirmação impressa do processo, que poderia ser feita através da geração de uma pulseira ou de uma ficha de encaminhamento, conforme mostra a Figura 12.

Figura 12 - Resultado da geração de arquivo de PDF para *download*, após confirmação do *check-in*

Fonte: Autoria própria.

A rota de Encaminhamento, identificada por "/referral", é de uso exclusivo do enfermeiro e lista os procedimentos médicos por hora e os pacientes com *check-in* realizado, conforme apresenta o Quadro 7.

Quadro 7 - Relação das rotas de Encaminhamento disponibilizadas pelo servidor

Rotas de Encaminhamento		
<i>Endpoint</i>	Método HTTP	Função
/referral/search	POST	Recebe um intervalo de horas e retorna todos os procedimento médicos que ocorrerão neste intervalo
/referral/referralBy Checkin	POST	Recebe um intervalo de horas e retorna todos os pacientes com <i>check-in</i> realizado neste intervalo

Fonte: Autoria própria.

A rota de Exames, identificada por "/exames", permite o *upload*, listagem, *download* e exclusão de arquivos relacionados aos procedimentos médicos, facilitando o registro de exames dos pacientes. O detalhamento das rotas mencionadas é feito no Quadro 8.

Quadro 8 - Relação das rotas de Exames disponibilizadas pelo servidor

Rotas de Exames		
<i>Endpoint</i>	Método HTTP	Função
/exames/upload	POST	Realiza o <i>upload</i> de arquivos no Google Drive
/exames/download/:fileId/:fileName	GET	Realiza o <i>download</i> do arquivo presente no Google Drive que possui as identificações de valores passados pelos parâmetros 'fileId' e 'fileName'
/exames/list/:folder	GET	Lista todos os arquivos de uma pasta do Google Drive que possui o valor do parâmetro de 'folder'
/exames/remove/:fileId	DELETE	Remove o arquivo do Google Drive que possui a identificação do valor passado no parâmetro 'fileId'

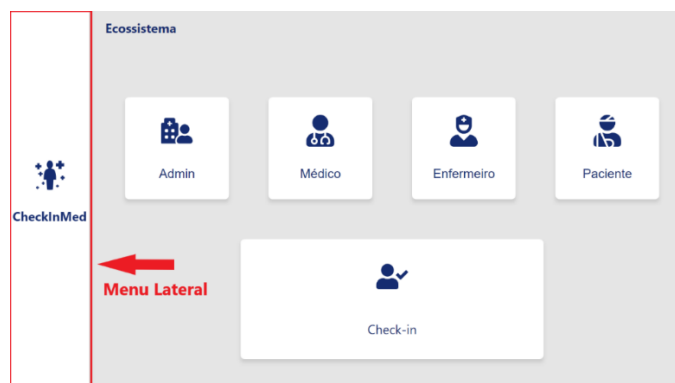
Fonte: Autoria própria.

4.3 Páginas construídas para o sistema

4.3.1 Menus laterais utilizados nas páginas do sistema

Foram construídos dois menus laterais com base no modelo definido no Figma, conforme descreve a seção 3.1.2.1. O primeiro é um menu que exibe apenas o logo do sistema que, ao ser clicado, redireciona para a página inicial (*Home*). Este menu foi projetado para ser apresentado apenas nas páginas de *Home* e *Check-in* e pode ser visualizado na Figura 13.

Figura 13 - Menu lateral construído para ser utilizado nas páginas de *Home* e *Check-in*



Fonte: Autoria própria.

O segundo menu possui os componentes de um menu básico. Ele exibe um logo pequeno que redireciona para a página inicial (*Home*), exibe o tipo de usuário logado e apresenta opções de redirecionamento. Estas opções são filtradas de acordo com o tipo de usuário logado: o nível Admin tem acesso às opções Usuários e Procedimentos; os níveis Médico e Paciente têm acesso às opções Procedimentos e Perfil; o nível Enfermeiro tem acesso às opções Procedimentos, Perfil e Encaminhamento. Este menu é utilizado quando se escolhe uma das opções de usuário na tela inicial (*Home*) e pode ser visualizado na Figura 14.

Figura 14 - Menu lateral fornecido para o tipo de usuário Enfermeiro e que também utilizado para os demais tipos de usuários

The screenshot shows the 'Procedimentos' page. On the left, a sidebar menu is visible with the following items: 'CheckInMed' (logo), 'Enfermeiro' (user profile), 'Procedimentos', 'Perfil', and 'Encaminhamento'. The main content area features a table with the following data:

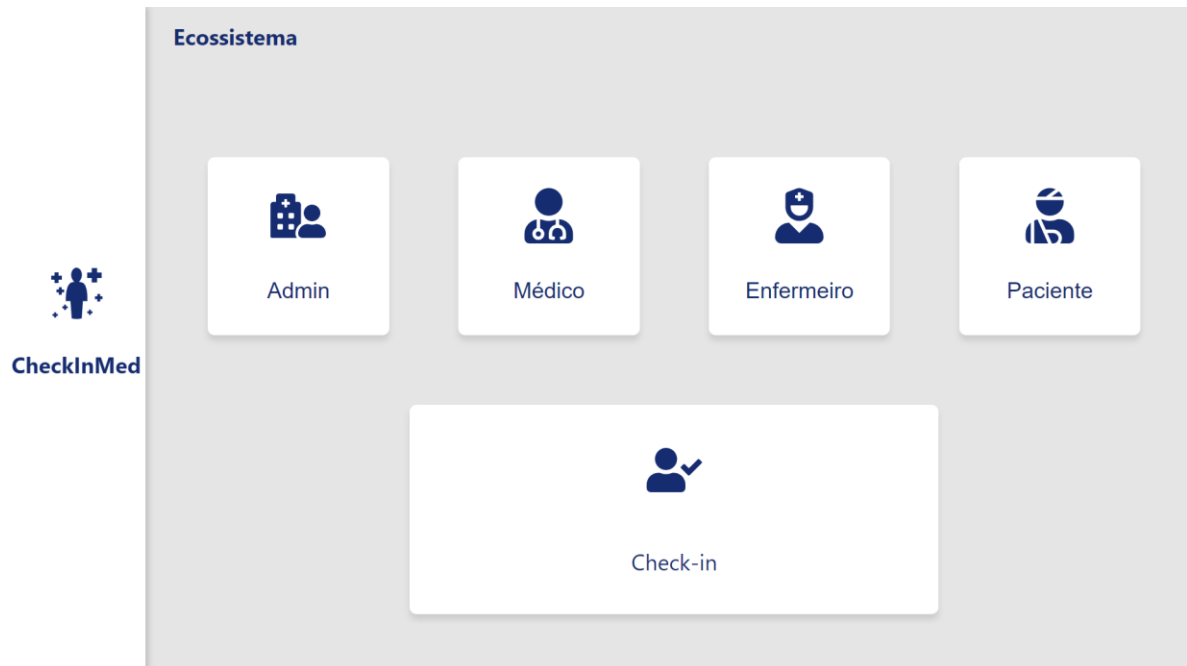
Paciente	Procedimento	Data	Medico Responsável	Check-in	Hr. Check-in	Ação
Sandra Sara da Paz	Tomografia Computadorizada	26/03/2024	Anderson Victor Yago Farias	Ausente		
Renato Cláudio Caldeira	Cirurgia de Coluna	12/03/2024	Anderson Victor Yago Farias	Ausente		
Rafael Breno Cavalcanti	Cirurgia de Emergência	29/04/2024	Daniel Filipe Luiz Oliveira	Ausente		

Fonte: Autoria própria.

4.3.2 Página inicial (*Home*)

A página inicial da aplicação *front-end* foi construída de forma que os usuários encontrassem cinco opções de *cards*, ou seja, representações visuais dos tipos de acesso dos usuários. Um dos *cards* foi criado com o intuito de permitir o acesso ao fluxo de *check-in* enquanto os outros quatro, dispostos lado a lado, representam os diferentes tipos de usuário: Admin, Médico, Enfermeiro e Paciente. Cada *card* é identificado por um ícone correspondente ao tipo de acesso, facilitando a identificação visual, conforme mostra a Figura 15.

Figura 15 - Página Inicial com o menu contendo o logo do sistema e os *cards* que dão acesso às páginas do sistema



Fonte: Autoria própria.

Foi desenvolvido um modal para que o usuário pudesse inserir o CPF e validar o acesso ao fluxo desejado após clicar em qualquer um dos *cards* de tipos de usuários.

É válido ressaltar que o acesso a página de *Check-in* foi inserido na tela inicial para os testes do protótipo, contudo, em uma implementação efetiva do produto final, não seria possível acessá-la a partir do *Home*, já que ela seria um acessório adicional integrado a um totem com impressora.

4.3.3 Página de *check-in*

A página de *Check-in* foi desenvolvida para simplificar o processo de registro de pacientes para procedimentos médicos. Na tela, foram criados um campo para inserir o CPF do paciente e um botão para enviar esse CPF para o *back-end*. Abaixo deles, inseriu-se um quadro que exibe o status do retorno do *back-end* após o envio do CPF, conforme mostra a Figura 16.

Figura 16 - Página de *Check-in* apresentando o estado inicial, com a instrução de inserir o CPF

A imagem mostra a interface de usuário para o processo de check-in. No topo, há um campo de entrada rotulado 'CPF' com o valor '000.000.000-00' e um ícone de seta para a direita. Abaixo do campo, um balão de mensagem centralizado contém o texto 'Por favor, insira seu CPF acima.'. À esquerda, há um ícone de uma pessoa com pontos ao redor e o texto 'CheckInMed'.

Fonte: Autoria própria.

Esse quadro pode assumir cinco estados distintos:

- **Estado Inicial:** indica onde o CPF pode ser inserido, conforme mostra a Figura 16.
- **Erro ou Procedimento Não Encontrado:** Mostra um ícone vermelho e informa o usuário para chamar um responsável caso ocorra algum erro ou não seja encontrado o procedimento do dia, conforme mostra a Figura 17.

Figura 17 - Página de *Check-in* com a apresentação do estado de erro, com a instrução que não há procedimento para o paciente

A imagem mostra a interface de usuário para o processo de check-in em um estado de erro. No topo, há um campo de entrada rotulado 'CPF' com o valor '008.709.188-79' e um ícone de seta para a direita. Abaixo do campo, um balão de mensagem centralizado contém o texto 'Aviso!' em vermelho, seguido de um ícone de uma cruz vermelha e o texto 'Nenhum procedimento registrado para hoje. Por favor, contate um responsável.'. À esquerda, há um ícone de uma pessoa com pontos ao redor e o texto 'CheckInMed'.

Fonte: Autoria própria.

• **Informações do Paciente e do Procedimento:** Exibe as informações do paciente e um breve detalhe do procedimento. Nesse estado, um botão para confirmar o *check-in* é apresentado. Este estado é exibido na Figura 18.

Figura 18 - Página de *Check-in* com procedimento encontrado na base de dados, contendo as informações do paciente, do procedimento e o botão de *Check-in*

The screenshot displays the 'CheckInMed' interface. At the top, there is a search bar with the CPF '008.709.188-77' and a right-pointing arrow. Below this is a section titled 'Verifique seus dados'. The main content is divided into three expandable sections: 'Informações Gerais', 'Endereço', and 'Procedimento'. Each section contains a table of data.

Informações Gerais					
Nome Completo	Data de nascimento	Nome da mãe	E-mail		
Aurora Jessica Rosângela Fernandes	1987-02-04T00:00:00.000Z	Jaqueline Alice Stefany	aurora_jessica_fernandes@santofoneira.ato.br		
CPF	RG	Telefone	Telefone de emergência	Sexo	
008.709.188-77	39.529.747-3	(15) 93872-5432	(15) 93872-5432	Feminino	

Endereço			
Rua	Número	Complemento	CEP
Rua Izabel Maria Rodrigues	100		18078-130
Cidade	Estado		
Sorocaba	São Paulo		

Procedimento			
Nome do procedimento	Médico responsável	Data	Hora
Cirurgia Cardíaca	Allana Louise Oliva Gonçalves	07/04/2024	19:45

At the bottom right of the form is a blue button labeled 'CHECK-IN'. On the left side of the page, there is a logo for 'CheckInMed' featuring a person icon with a plus sign.

Fonte: Autoria própria.

• **Check-in Confirmado:** Mostra um ícone verde e uma mensagem informando que o *check-in* foi realizado com sucesso, conforme mostra a Figura 19. Após a confirmação do *check-in*, um PDF com as informações do paciente é baixado automaticamente.

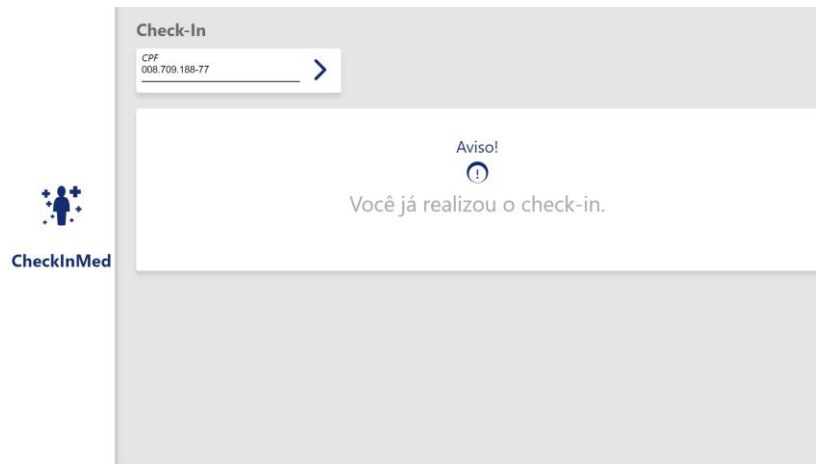
Figura 19 - Página de *Check-in* com apresentação de mensagem de *check-in* bem-sucedido

The screenshot shows the 'Check-In' page after a successful check-in. At the top, the search bar still shows the CPF '008.709.188-77'. The main area is dominated by a large white box with a green checkmark icon and the text 'Aviso! Check-in realizado com sucesso!'. The 'CheckInMed' logo is visible on the left side.

Fonte: Autoria própria.

- **Check-in Já Realizado:** Exibe um ícone azul de aviso informando que o *check-in* já foi realizado para o mesmo procedimento. Isso ocorre ao clicar no botão de *check-in* novamente para o mesmo procedimento, conforme mostra a Figura 20.

Figura 20 - Página de *Check-in* com apresentação da mensagem informando que o *check-in* já foi realizado



Fonte: Autoria própria.

4.3.4 Páginas de listagem e detalhes de usuário

A página de listagem de Usuários é exclusiva para o Administrador e foi projetada para fornecer uma visão geral dos usuários cadastrados no sistema, conforme mostra a Figura 21. Na tela, há um botão de pesquisa e abaixo dele, uma tabela que exibe algumas informações dos usuários retornados. Os dados são carregados na tabela ao acessar a página ou ao clicar no botão de pesquisa. Um destaque importante é o último item em cada linha da tabela, que consiste em um ícone de redirecionamento para a página de detalhes daquele usuário, fornecendo o ID presente em cada item da tabela.

Figura 21 - Página de listagem de usuários na visão de um Admin

Nome	Tipo	Documento	Telefone	Ação
Admin	Admin	135.421.528-11		
Anderson Victor Yago Farias	Médico(a)	866.565.468-20	(15) 98104-2018	
Daniel Filipe Luiz Oliveira	Médico(a)	744.074.268-02	(15) 99374-5900	
Ailana Louise Otília Gonçalves	Médico(a)	027.204.288-11	(15) 99589-0415	
Marina Kamilly Melissa Lopes	Enfermeiro(a)	530.657.328-23	(19) 3815-7046	
Rosângela Sara Jacqueline Galvão	Enfermeiro(a)	046.480.068-44	(11) 3896-2825	
Sarah Mariana Rocha	Enfermeiro(a)	573.981.978-45	(19) 2918-6503	
Samuel Benedito Henry Cavalcanti	Enfermeiro(a)	950.867.888-70	(11) 3795-4573	
Isabelle Liz Gabrielly da Luz	Enfermeiro(a)	019.853.288-10	(19) 3761-0239	
Elza Andréia Andrea da Cruz	Paciente	165.254.318-08	(15) 92804-9300	

Fonte: Autoria própria.

A página de detalhes do Usuário, exibida na Figura 22, pode ser acessada pela tabela de usuários ou pela opção de menu "Perfil". Ela possui dois componentes expansíveis: "Informações Gerais" e "Endereço", além de um botão de salvar no final da página. O componente "Informações Gerais" exibe todas as informações do usuário, sendo que dois campos têm um controle diferente: o "COREN" é mostrado apenas se o usuário for do tipo enfermeiro, enquanto o "CRM" é exibido apenas se o usuário for do tipo médico. Isso ocorre porque o "COREN" corresponde ao registro do enfermeiro, e o "CRM" refere-se ao registro do médico. Já o componente "Endereço" é exibido apenas se o usuário for do tipo paciente e apresenta as informações básicas de endereço. Todos os campos nesta tela são editáveis e, ao clicar no botão de salvar, as alterações são salvas.

Figura 22 - Página de detalhe de um usuário do tipo Paciente

CheckinMed

Admin
exemplo@exemplo.com

- Usuários
- Procedimentos

Informações do usuário

Informações Gerais

Nome Completo	Data de nascimento	Nome da mãe	E-mail
Elza Andrea Andrea da Cruz	1996-02-13T00:00:00.000Z	Yasmin Fabiana Tallane	elza.andrea.dacruz@gmail.com
CPF	RG	Telefone	Telefone de emergência
165.254.319-08	17.400.697-4	(15) 92804-9200	(15) 92804-9200
			Sexo
			Feminino

Endereço

Rua	Número	Complemento	CEP
Praça Maria Cristina Lolatta Pereira	857		18051-595
Cidade	Estado		
Sorocaba	São Paulo		

SALVAR

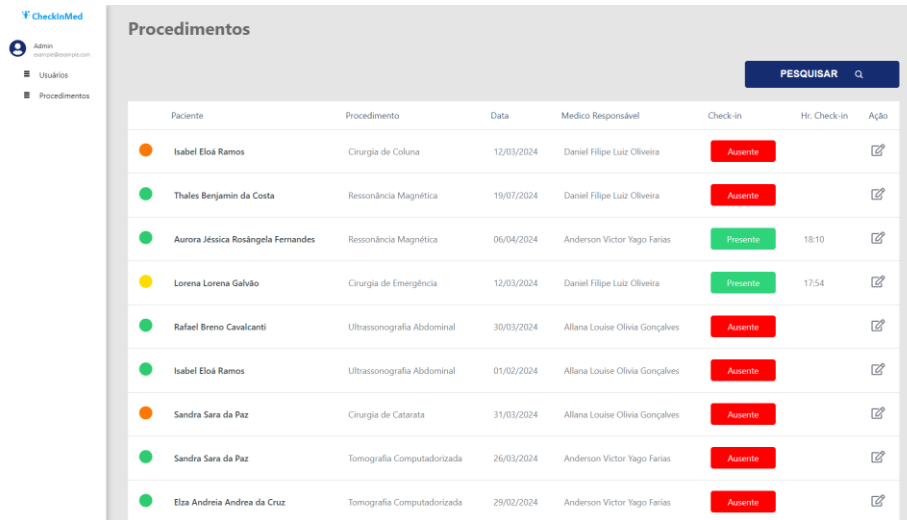
Fonte: Autoria própria.

4.3.5 Páginas de listagem e detalhes de procedimentos

A página de listagem de Procedimentos foi desenvolvida para fornecer uma visão geral dos procedimentos cadastrados no sistema. Nela, encontra-se um botão de pesquisa e, abaixo dele, uma tabela que exibe algumas informações dos procedimentos retornados, conforme mostra a Figura 23. Os dados na tabela são carregados ao acessar a página ou clicando no botão de pesquisa.

Além disso, cada linha da tabela possui um item final, representado por um ícone que possui comportamento de redirecionamento para a página de detalhes daquele procedimento. Esse redirecionamento é feito com base no ID presente em cada item da tabela.

Figura 23 - Página de listagem de procedimentos na visão de um usuário do tipo Admin



Paciente	Procedimento	Data	Medico Responsável	Check-in	Hr. Check-in	Ação
Isabel Eloá Ramos	Cirurgia de Coluna	12/03/2024	Daniel Filipe Luiz Oliveira	Ausente		
Thales Benjamin da Costa	Ressonância Magnética	19/07/2024	Daniel Filipe Luiz Oliveira	Ausente		
Aurora Jéssica Rosângela Fernandes	Ressonância Magnética	06/04/2024	Anderson Victor Yago Farias	Presente	18:10	
Lorena Lorena Galvão	Cirurgia de Emergência	12/03/2024	Daniel Filipe Luiz Oliveira	Presente	17:54	
Rafael Breno Cavalcanti	Ultrassonografia Abdominal	30/03/2024	Allana Louise Olívia Gonçalves	Ausente		
Isabel Eloá Ramos	Ultrassonografia Abdominal	01/02/2024	Allana Louise Olívia Gonçalves	Ausente		
Sandra Sara da Paz	Cirurgia de Catarata	31/03/2024	Allana Louise Olívia Gonçalves	Ausente		
Sandra Sara da Paz	Tomografia Computadorizada	26/03/2024	Anderson Victor Yago Farias	Ausente		
Eiza Andréia Andrea da Cruz	Tomografia Computadorizada	29/02/2024	Anderson Victor Yago Farias	Ausente		

Fonte: Autoria própria.

A página de detalhe de Procedimento é acessada exclusivamente através da tabela de procedimentos. Essa página é composta por sete componentes expansíveis, sendo que cinco deles aparecem para todos os tipos de usuários e dois deles (prontuários) aparecem apenas para alguns tipos de usuários. Cada um dos componentes é descrito a seguir:

4.3.5.1 Componente 1 – Informações Gerais

Este componente apresenta informações sobre as partes envolvidas no procedimento, como nome do paciente, médico responsável, enfermeiro responsável, nome, dia e horário do procedimento, horário e status do *check-in* realizado, conforme mostra a Figura 24.

Figura 24 - Página de detalhes do procedimento médico com enfoque nas Informações Gerais



Informações Gerais					
Paciente	Médico Responsável		Enfermeiro responsável		
Lorena Lorena Galvão	Daniel Filipe Luiz Oliveira		Samuel Benedito Henry Cavalcanti		
Nome do procedimento	Data	Horário	Hr. check-in	Check-in	
Cirurgia de Emergência	12/03/2024	18:30	17:54	Presente	
Exames					
Consultas					
Histórico médico					
Prontuário - Enfermeiro					
Prontuário - Médico					
Prescrição Médica					

Fonte: Autoria própria.

4.3.5.2 Componente 2 – Exames

Neste componente, exibido na Figura 25, é realizada a integração com o Google Drive. Aqui são listados os arquivos presentes na pasta compartilhada com o procedimento. Ao clicar em um item da lista, o arquivo é baixado. Além disso, há um botão para realizar o *upload* de 1 ou mais arquivos (no formato PDF ou JPG) no Google Drive. Após o *upload*, a lista de arquivos é atualizada após 5 segundos.

Figura 25 - Página de detalhes do procedimento médico com enfoque nos Exames

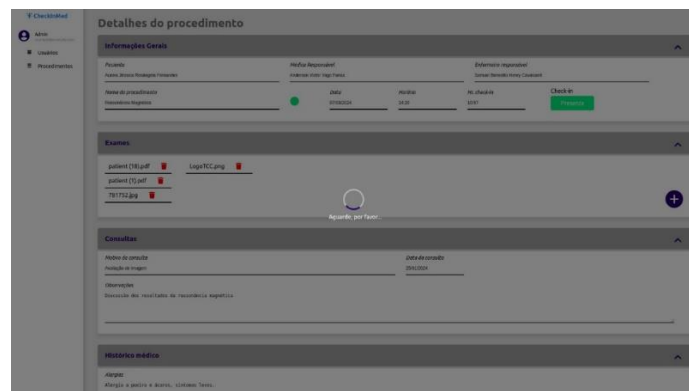


Fonte: Autoria própria.

Outra funcionalidade adicionada foi a de um ícone de remoção para cada arquivo listado, permitindo que os usuários excluam arquivos facilmente. Tal funcionalidade também possui o mesmo tempo de espera que o *upload* para ser atualizada.

A fim de apresentar um melhor comportamento para estas funcionalidades, durante o tempo de espera, foi implementada uma tela de carregamento para indicar que a execução da operação, visto na Figura 26. Isso foi importante, pois a comunicação com o Google Drive possui um atraso considerável para ser propagada.

Figura 26 - Comportamento de carregamento ao fazer o upload de arquivo



Fonte: Autoria própria.

4.3.5.3 Componente 3 – Consultas

Este item, apresentado na Figura 27, mostra os registros de consultas anteriores pelas quais o paciente passou antes do procedimento em questão, disponíveis na base de dados do hospital ou clínica.

Figura 27 - Página de detalhes do procedimento médico com enfoque nas Consultas

The screenshot shows the 'CheckinMed' interface. On the left is a sidebar with a user profile 'Admin' and navigation options for 'Usuários' and 'Procedimentos'. The main content area is titled 'Detalhes do procedimento' and contains several expandable sections: 'Informações Gerais', 'Exames', 'Consultas', 'Histórico médico', 'Prontuário - Enfermeiro', 'Prontuário - Médico', and 'Prescrição Médica'. The 'Consultas' section is currently expanded, displaying a table with columns for 'Motivo da consulta' and 'Data da consulta'. The table contains one entry: 'Consulta pré-cirúrgica' on '25/01/2024'. Below the table is a text area for 'Observações' with the content 'Avaliação antes da cirurgia de joelho'.

Fonte: Autoria própria.

4.3.5.4 Componente 4 – Histórico Médico

Este componente, apresentado na Figura 28, está relacionado ao paciente e serve como registro de alergias, cirurgias e doenças anteriores.

Figura 28 - Página de detalhes do procedimento médico com enfoque no Histórico Médico

The screenshot shows the 'CheckinMed' interface with the 'Histórico médico' section expanded. The 'Alergias' section contains the text 'Alergia a certos tipos de frutos do mar, sistemas jeans.'. The 'Histórico de doenças' section contains the text 'Histórico de alergias sazonais desde a infância. Sofreu uma lesão no joelho aos 28 anos, exigindo cirurgia de reconstrução do ligamento cruzado anterior (LCA)'. The 'Cirurgias anteriores' section contains the text 'Nenhuma cirurgia anterior além da reconstrução do LCA.'.

Fonte: Autoria própria.

4.3.5.5 Componente 5 – Prontuário (Enfermeiro)

Este componente (Figura 29) é editável e é destinado ao registro do que foi feito pelo enfermeiro no procedimento. Pode ser acessado pelo enfermeiro ou pelo administrador. Pacientes e médicos não têm acesso a ele.

Figura 29 - Página de detalhes do procedimento médico com enfoque no Prontuário do enfermeiro



Fonte: Autoria própria.

4.3.5.6 Componente 6 – Prontuário (Médico)

Semelhante ao componente anterior, este (Figura 30) é destinado ao registro do que foi feito pelo médico no procedimento. Pode ser acessado pelo médico ou pelo administrador, mas não pelos pacientes ou enfermeiros.

Figura 30 - Página de detalhes do procedimento médico com enfoque no Prontuário do médico



Fonte: Autoria própria.

4.3.5.7 Componente 7 – Prescrição Médica

Este é o último componente da página e é editável (Figura 31). Aqui, apenas o médico pode preencher informações sobre medicamentos, incluindo nome, quantidade da dose e instrução de uso. O médico também pode remover itens da prescrição e gerar o arquivo para impressão da receita. Porém, para outros tipos de usuários, é um campo destinado apenas para visualização.

Para a remoção de algum item presente na prescrição médica, os elementos foram identificados com números, ou índices. Tais identificações podem ser referenciadas no ícone de lixeira, no canto superior direito da área da prescrição médica, já que, ao se clicar em tal ícone, um modal é aberto solicitando o número do índice que se deseja excluir. Após a confirmação, a exclusão é efetuada.

Já para a geração de arquivo, adicionou-se um ícone que remete à ideia de um PDF e se encontra ao lado do ícone de lixeira. Esse ícone tem como função permitir que os médicos visualizem o PDF do receituário gerado, pois ao clicar nele, o PDF é aberto em uma nova aba do navegador, facilitando a visualização e a revisão por parte do profissional de saúde.

É importante ressaltar que qualquer alteração feita na prescrição médica deve ser salva adequadamente no sistema. Isso garante que o arquivo PDF gerado reflita sempre as informações mais atualizadas.

Figura 31 - Página de detalhes do procedimento médico com enfoque na Prescrição Médica

CheckinMed

Médico
medico@exemplo.com

Procedimentos

Perfil

Informações Gerais

Exames

Consultas

Histórico médico

Prontuário - Médico

Prescrição Médica

PDF

1) Clprofloxacino X 500mg - Tomar 1 comprimido a cada 12 horas por 7 dias
2) teste X teste - teste

Medicamento Dose Instruções

SALVAR

Fonte: Autoria própria.

No final da página, há um botão para salvar as alterações feitas nos componentes 5, 6 e 7. Já os *uploads* de arquivos são salvos imediatamente no momento do envio.

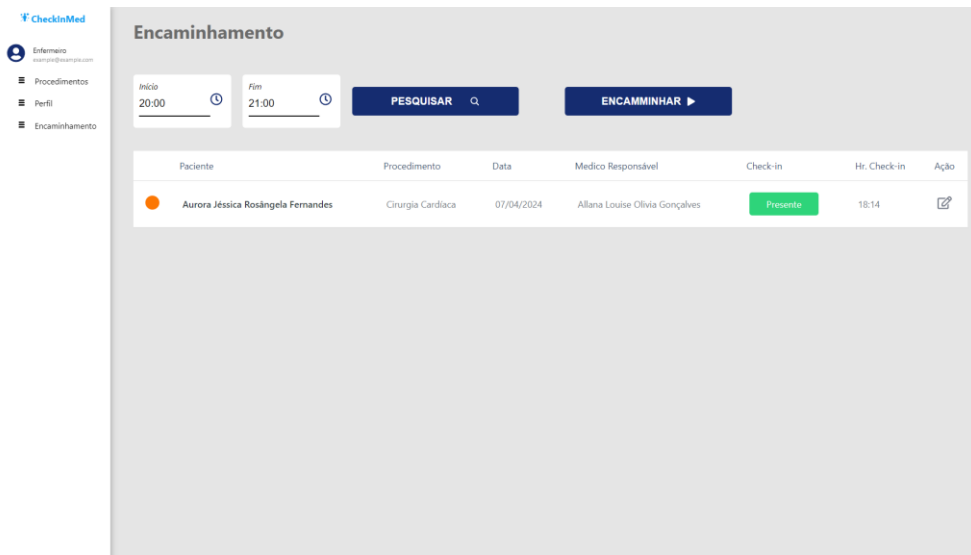
4.3.6 Página de encaminhamento

A página de Encaminhamento é exclusiva para uso do usuário do tipo Enfermeiro, apresenta uma funcionalidade específica para auxiliar na organização dos procedimentos do dia e possui o mesmo componente de tabela presente na página de Procedimentos, conforme mostra a Figura 32.

Acima da tabela, há dois componentes para inserir os limites do intervalo de tempo em que é possível filtrar os procedimentos do dia relacionados ao enfermeiro responsável. Ao clicar no botão "Pesquisar", os dados são retornados e preenchem a tabela.

Quando há dados para preencher a tabela, é habilitado um botão "Encaminhar". Ao clicar neste botão, é iniciada uma automação com as horas selecionadas.

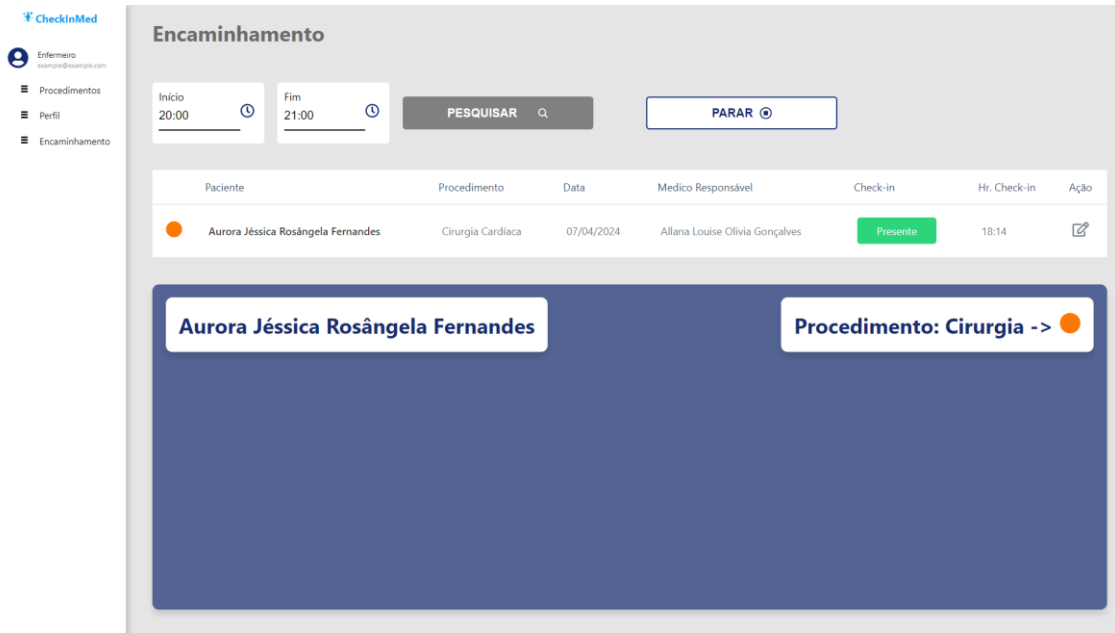
Figura 32 - Página de encaminhamento com a definição de horário, os botões de ação (Pesquisar e Encaminhar) e a tabela dos procedimentos filtrados



Fonte: Autoria própria.

Para visualizar e simular a chamada de um painel, foi construído um quadro na própria tela de Encaminhamento, mostrando o nome do paciente, tipo do procedimento e a cor que remete à área que o procedimento será realizado, conforme apresenta a Figura 33. Os dados a serem mostrados neste quadro são atualizados a cada 1 minuto, conforme a automação é inicializada.

Figura 33 - Página de encaminhamento em execução (após clicar no botão ENCAMINHAR) destacando os nomes a serem chamados para os respectivos procedimentos médicos



Fonte: Autoria própria.

Caso seja necessário encerrar o processo manualmente, o botão "Encaminhar" se transforma em "Parar". Ao clicar neste botão, as funções de *setInterval* e *setTimeout* são interrompidas. Isso é feito executando as funções *clearInterval* e *clearTimeout*, respectivamente, e passando os IDs associados, que estão salvos no *sessionStorage*, como parâmetros para as mesmas.

Após a interrupção dessas funções, todos os outros dados salvos no *sessionStorage* são limpos e a página é resetada para iniciar um novo processo, garantindo que o usuário possa iniciar novamente a automação ou ajustar as configurações de tempo conforme necessário.

As páginas apresentadas anteriormente podem ser visualizadas por meio do *link* de *deploy* da aplicação contido no apêndice F.

4.4 Apresentação e discussão das respostas do formulário de validação do sistema

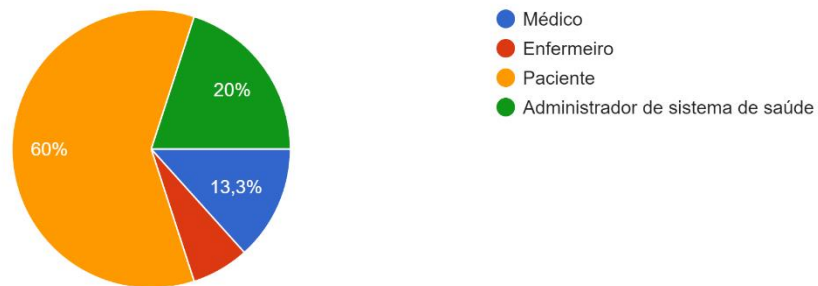
Nesta seção, as respostas coletadas no formulário de validação do sistema são apresentadas e discutidas. Com o intuito de auxiliar a sua análise, esta seção está dividida em três tópicos: funcionalidade e usabilidade; inovação; satisfação e aceitação.

4.4.1 Caracterização dos participantes do processo de validação

Averiguou-se a área de atuação em que os entrevistados se enquadravam, conforme exibe o gráfico da Figura 34.

Figura 34 - Gráfico de avaliação dos tipos de usuários em que os participantes se enquadram

Em qual tipo de usuário você se enquadra?
15 respostas



Fonte: Autoria própria.

Analisando-se o gráfico da Figura 34, percebe-se que 40% dos entrevistados atuam na área da saúde. Contudo, esse percentual não se reflete necessariamente nas demais respostas coletadas, já que os usuários, mesmo que instruídos a responder prioritariamente o fluxo a que pertenciam, tiveram a liberdade de testar outros tipos de usuário.

4.4.2 Funcionalidade e usabilidade

Esta seção visou avaliar as funcionalidades e a usabilidade para os diferentes tipos de usuários do sistema, considerando os pontos avaliados sobre os recursos oferecidos, experiência do usuário e clareza das informações. Além disso, foram coletadas informações referentes à problemas encontrados e sugestões de melhorias para o sistema.

4.4.2.1 Fluxo de Admin

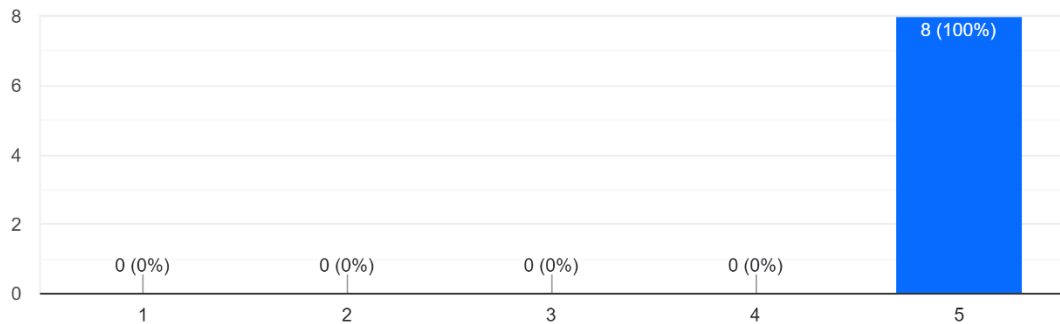
Para o Fluxo de Admin, foram obtidas 8 respostas, sendo 3 de pessoas que se categorizaram como administradores da área da saúde.

As respostas pertinentes à intuitividade e facilidade de uso são exibidas na Figura 35.

Figura 35 - Gráfico de avaliação da intuitividade e facilidade de uso do Fluxo de Admin

O quanto o Fluxo de Admin é intuitivo e fácil de usar?

8 respostas



Fonte: Autoria própria.

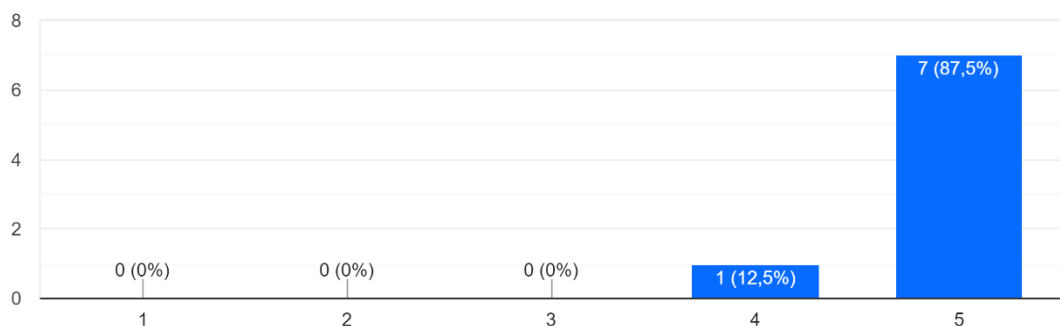
Analisando-se o gráfico da Figura 35, nota-se que nenhum dos participantes teve dificuldades em mexer ou consultar as informações fornecidas para o usuário.

Já no quesito de abrangência das funcionalidades do fluxo, não houve unanimidade entre todos os entrevistados, conforme mostra a Figura 36, porém todas as avaliações foram positivas (notas 4 ou 5), indicando que para as ações concedidas ao administrador estão boas ou excelentes.

Figura 36 - Gráfico de avaliação da abrangência das funcionalidades do Fluxo de Admin

Considerando o Fluxo de Admin, como você avalia a abrangência de suas funcionalidades?

8 respostas

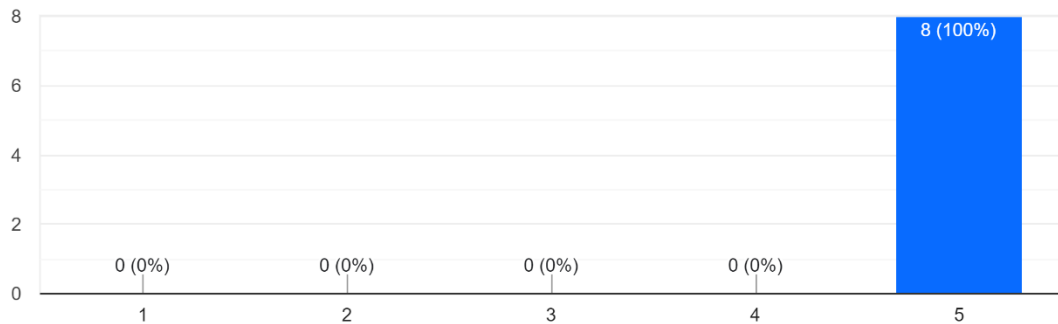


Fonte: Autoria própria.

Assim como os gráficos anteriores, o gráfico da Figura 37 demonstra que a qualidade das funcionalidades do fluxo de administrador foram muito bem avaliadas pelos participantes.

Figura 37 - Gráfico de avaliação da qualidade das funcionalidades do Fluxo de Admin

Considerando o Fluxo de Admin, como você avalia a qualidade de suas funcionalidades?
8 respostas



Fonte: Autoria própria.

Os comentários de sugestões de melhoria para o fluxo de administrador podem ser vistos no apêndice G. Dentre estes comentários, destacam-se dois que trazem a sugestão de limitar as ações do administrador, impedindo que ele tenha acesso aos exames, e também de fazer com que a página de detalhe do procedimento esteja disponível exclusivamente para fins de consulta.

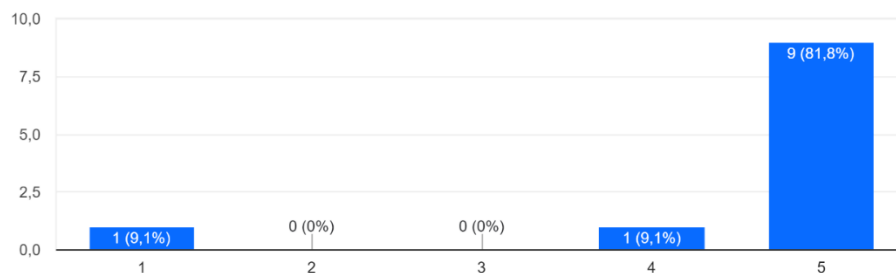
4.4.2.2 Fluxo de Médico

Para o Fluxo de Médico, foram obtidas 11 respostas, sendo 2 de pessoas que se categorizaram como médicos.

As respostas pertinentes à intuitividade e facilidade de uso são exibidas na Figura 38.

Figura 38 - Gráfico de avaliação da intuitividade e facilidade de uso do Fluxo de Médico

O quanto o Fluxo de Médico é intuitivo e fácil de usar?
11 respostas



Fonte: Autoria própria.

Observa-se na Figura 38, que a maioria dos participantes considerou o fluxo de médico muito fácil de se utilizar (nota 5), contudo, a avaliação igual a 1 contida no gráfico indica que uma pessoa considerou a utilização do fluxo em questão muito difícil. Ao se analisar o comentário referente à avaliação igual a 1, exibido na Figura 39, percebe-se que a dificuldade reportada é proveniente da exibição separada dos prontuários de médico e de enfermeiro, não de uma dificuldade técnica.

Figura 39 - Comentário referente à avaliação baixa da intuitividade e facilidade de uso do Fluxo de Médico

Apenas na parte de prontuário, onde ficaria de mais fácil funcionalidade para equipe médica os prontuários tanto medico como do enfermeiro ficassem em uma aba em comum, para melhor comunicação da equipe.

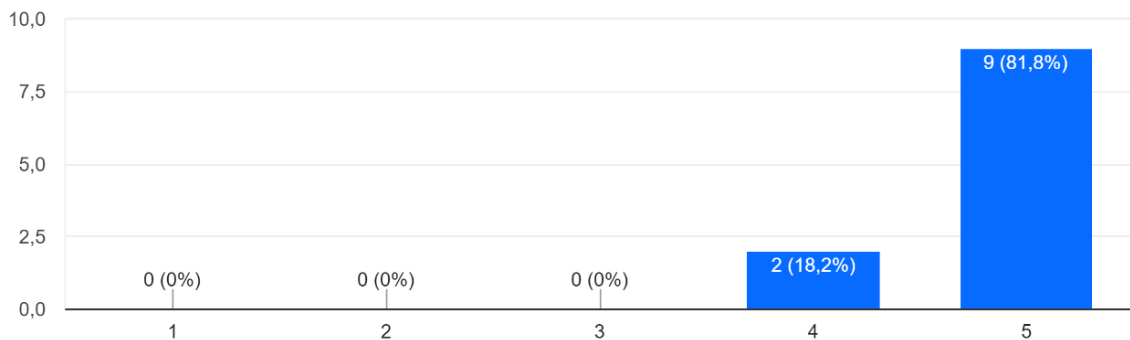
Fonte: Autoria própria.

Já no quesito de abrangência das funcionalidades do fluxo, não houve unanimidade entre todos os entrevistados, conforme mostra a Figura 40, porém todas as avaliações foram positivas (notas 4 ou 5), indicando que para as ações concedidas ao médico estão boas ou excelentes.

Figura 40 - Gráfico de avaliação da abrangência das funcionalidades do Fluxo de Médico

Considerando o Fluxo de Médico, como você avalia a abrangência de suas funcionalidades?

11 respostas

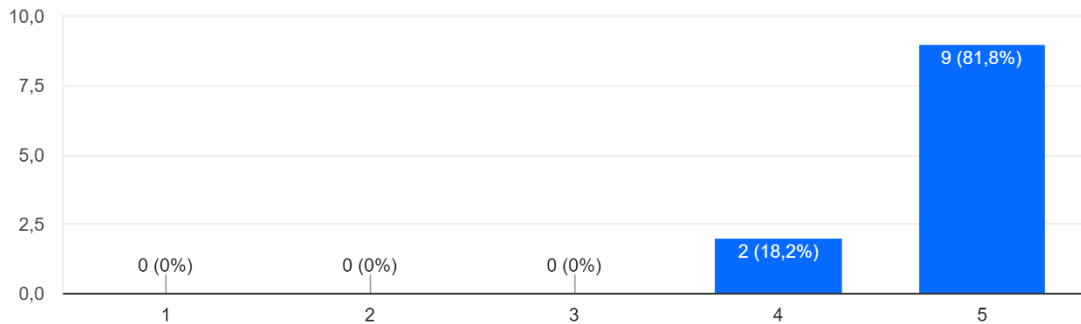


Fonte: Autoria própria.

Da mesma forma, observam-se avaliações positivas para as qualidades das funcionalidades do fluxo em questão, conforme mostra a Figura 41.

Figura 41 - Gráfico de avaliação da qualidade das funcionalidades do Fluxo de Médico

Considerando o Fluxo de Médico, como você avalia a qualidade de suas funcionalidades?
11 respostas



Fonte: Autoria própria.

Todos os comentários e sugestões de melhoria para o fluxo médico estão listados no apêndice H. Dentre esses comentários, destacam-se algumas sugestões relacionadas à prescrição médica. Uma delas sugere a inclusão de medicamentos comuns já existentes na base de dados do sistema, oferecendo a opção de selecioná-los em vez de digitá-los manualmente. Outra sugestão visa melhorar a disposição e a escrita dos medicamentos na tela, tornando a prescrição mais clara e de fácil entendimento. A última sugestão foi feita por um médico entrevistado e sugere que a receita médica seja corrigida para não exigir a explicação de cada informação contida na prescrição (por exemplo, não é necessário informar que '500mg' se refere à 'Dosagem').

Por fim, uma outra melhoria sugerida consistiu em avaliar as cores dos procedimentos para se enquadrarem em uma regra do SUS. De acordo com essa sugestão, as cores seriam atribuídas da mais grave para a mais branda, respectivamente: VERMELHA, AMARELA, VERDE e AZUL. Essas cores indicam a urgência do problema e o tempo máximo que o paciente pode esperar (NOVO HAMBURGO, 2020).

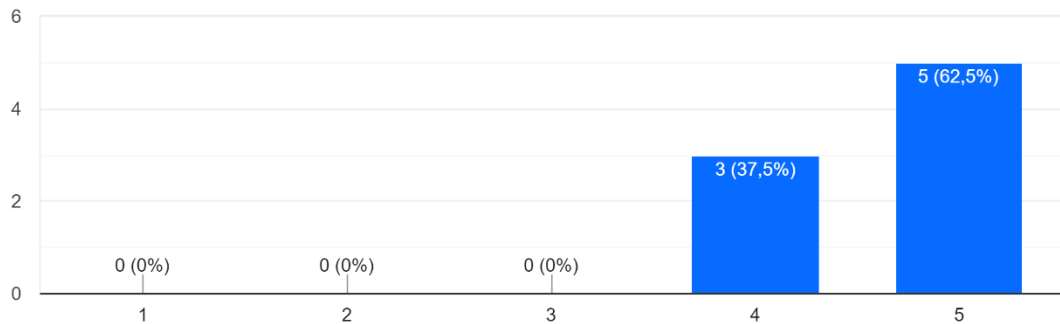
4.4.2.3 Fluxo de Enfermeiro

Para o Fluxo de Enfermeiro, foram obtidas 8 respostas, sendo 1 de um indivíduo que se classificou como enfermeiro.

O fluxo foi considerado intuitivo e fácil de usar pelos avaliadores, recebendo notas 4 e 5, conforme exhibe o gráfico da Figura 42.

Figura 42 - Gráfico de avaliação da intuitividade e facilidade de uso do Fluxo de Enfermeiro

O quanto o Fluxo de Enfermeiro é intuitivo e fácil de usar?
8 respostas

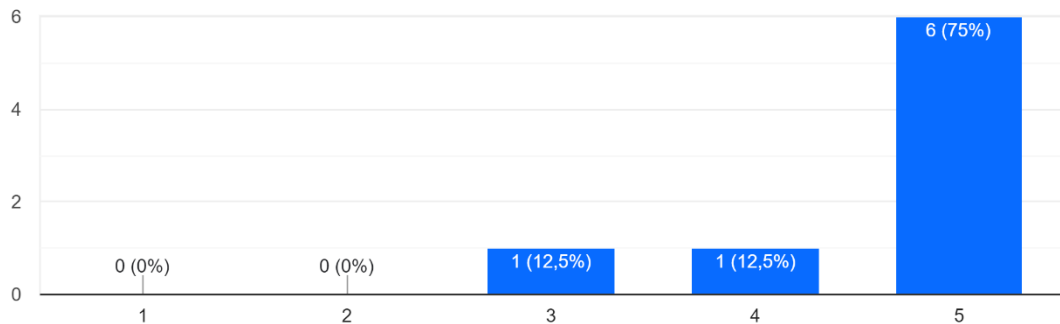


Fonte: Autoria própria.

O gráfico que avalia a abrangência das funcionalidades do fluxo de enfermeiro pode ser visualizado na Figura 43.

Figura 43 - Gráfico de avaliação da abrangência das funcionalidades do Fluxo de Enfermeiro

Considerando o Fluxo de Enfermeiro, o quão você avalia a abrangência de suas funcionalidades?
8 respostas



Fonte: Autoria própria.

Nota-se que, apesar da maioria das avaliações fornecerem nota máxima a este quesito, houve uma avaliação mediana (com nota 3). De acordo com o comentário pertinente a tal resposta, exibido na Figura 44, há uma deficiência no processo de encaminhamento, pois o mesmo permite que mais de um paciente com *check-in* realizado sejam encaminhados ao mesmo tempo para os preparativos dos procedimentos médicos, gerando conflito no atendimento.

Esta falha reflete a importância da variedade dos testes do protótipo bem como a inclusão de pessoas da área em questão na sua avaliação, já que, embora possa não afetar o encaminhamento quando não há muitos pacientes que fizeram *check-in* (conforme refletem as avaliações positivas obtidas), inviabiliza a implementação do protótipo em um ambiente hospitalar sem que tal funcionalidade seja corrigida.

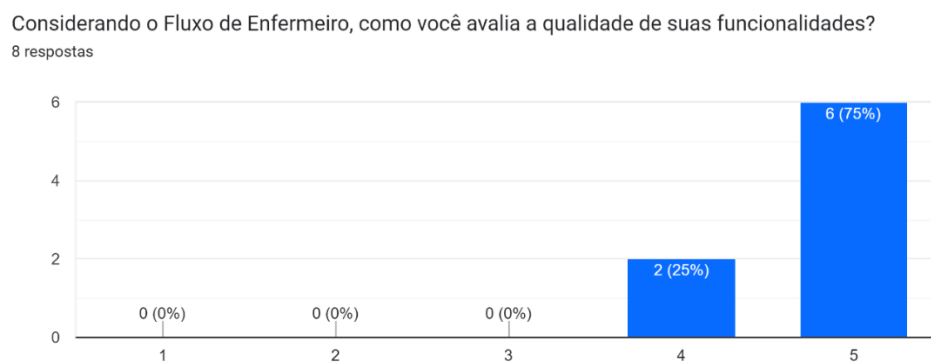
Figura 44 - Comentário referente à avaliação mediada da abrangência das funcionalidades do Fluxo de Enfermeiro

Sim, na parte encaminhamento/ triagem, qdo mais de um paciente fizer o check-in no intervalo de tempo, o sistema chamará vários pacientes ao mesmo tempo.

Fonte: Autoria própria.

Apesar da falha evidenciada anteriormente, a qualidade das funcionalidades de tal fluxo foi avaliada positivamente no gráfico da Figura 45, demonstrando que a ferramenta está bem estruturada e que é passível de ser implementada em uma aplicação prática após a correção do defeito evidenciado.

Figura 45 - Gráfico de avaliação da qualidade das funcionalidades do Fluxo de Enfermeiro



Fonte: Autoria própria.

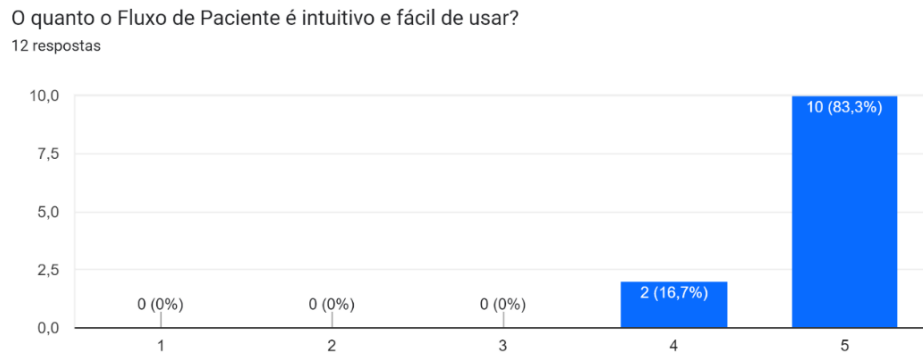
Os comentários e sugestões de melhoria para o fluxo de enfermeiro estão listados no apêndice I. Dentre esses comentários, destacam-se alguns relacionados ao processo de encaminhamento do paciente. Foi sugerido repensar a consulta por intervalo de horas, pois não foi considerado muito intuitivo. Além disso, sugere-se que o encaminhamento seja individualizado, permitindo a liberação de um botão em cada linha da tabela para quando o *check-in* estiver realizado, permitindo ao enfermeiro interromper esse processo conforme necessário. Outra melhoria mencionada foi disponibilizar o prontuário do paciente também para o enfermeiro, o que facilitaria a troca de informações entre enfermeiros e médicos.

4.4.2.4 Fluxo de Paciente

Para o Fluxo de Paciente, foram obtidas 12 respostas sendo 9 de pessoas que se enquadram exclusivamente como paciente.

As respostas pertinentes à intuitividade e facilidade de uso são exibidas na Figura 46.

Figura 46 - Gráfico de avaliação da intuitividade e facilidade de uso do Fluxo de Paciente

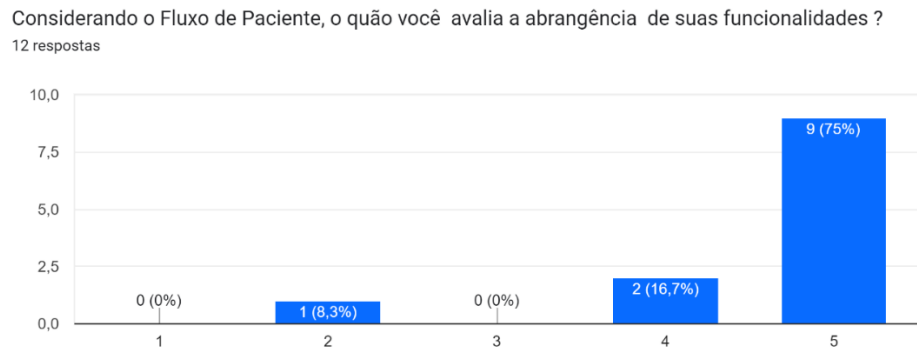


Fonte: Autoria própria.

Analisando-se o gráfico da Figura 46, nota-se que todos os participantes avaliaram a intuitividade e facilidade de uso como boa ou excelente para o fluxo de paciente.

Já no quesito de abrangência das funcionalidades do fluxo, é notada uma baixa avaliação no gráfico da Figura 47, indicando funcionalidades que não são ideais para o paciente. Ao verificar os comentários, em relação à resposta de grau 2, percebe-se um problema na consulta de informações. Já que ao disponibilizar dados médicos para todos os envolvidos no procedimento, pode ocorrer uma quebra do sigilo médico, pois o acesso às informações pode ser facilmente vazado. Esse comentário está registrado na Figura 48.

Figura 47 - Gráfico de avaliação da abrangência das funcionalidades do Fluxo de Paciente



Fonte: Autoria própria.

Figura 48 - Comentário referente à avaliação baixa da abrangência das funcionalidades do Fluxo de Paciente

Sim, ele só ter a opção de baixar seus exames, e apenas o médico poder incluir e excluir exames na tela de detalhe de procedimento. Ainda na tela de detalhe de procedimento, tirar tudo o que for relacionando a dado médico, deixar só informações do procedimento, a seção de "Informações gerais", pois deixando acesso a dados médicos pode ocorrer o caso de quebra de sigilo médico. A parte de check-in está muito boa.

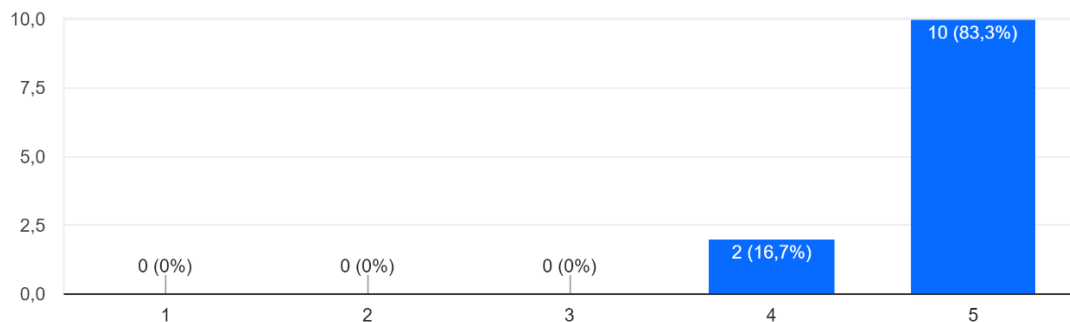
Fonte: Autoria própria.

Apesar do risco evidenciado anteriormente, a qualidade das funcionalidades de tal fluxo foi avaliada positivamente no gráfico da Figura 49.

Figura 49 - Gráfico de avaliação da qualidade das funcionalidades do Fluxo de Paciente

Considerando o Fluxo de Paciente, como você avalia a qualidade de suas funcionalidades?

12 respostas



Fonte: Autoria própria.

Os comentários e sugestões de melhoria para o fluxo de paciente estão disponíveis no apêndice J. Dentre esses comentários, destacam-se algumas sugestões relacionadas à página de detalhes do procedimento, como já mencionado anteriormente. Além disso, houve sugestão para o acesso dos exames, sugerindo que o paciente possa apenas baixar os arquivos, reservando a funcionalidade de carregar e excluir arquivos exclusivamente ao médico.

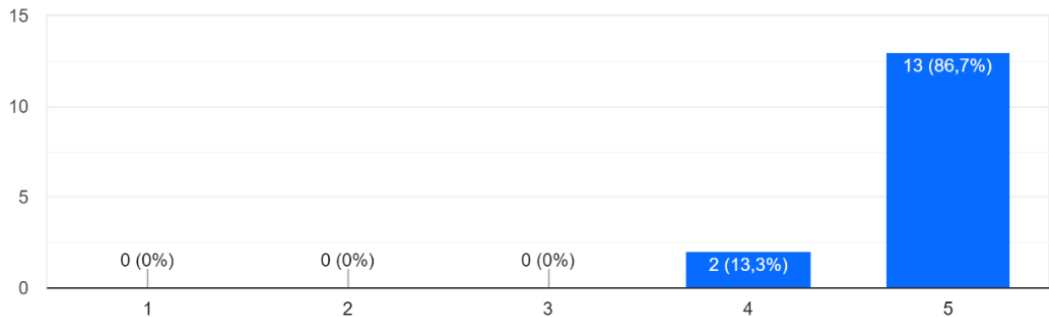
4.4.3 Inovação

Esta seção tem como objetivo avaliar o nível de inovação (isto é, algo novo que traz uma grande mudança) do sistema na área da saúde. Para tal avaliação, todos os entrevistados forneceram suas respostas analisando o sistema como um todo.

O grau de inovação obtido pelos participantes é exibido no gráfico da Figura 50, o qual indica que, para eles, o conceito do sistema consistiu em uma ideia nova para o sistema de saúde.

Figura 50 - Gráfico da avaliação do grau de inovação do sistema

Por favor, avalie o grau de inovação do produto apresentado.
15 respostas

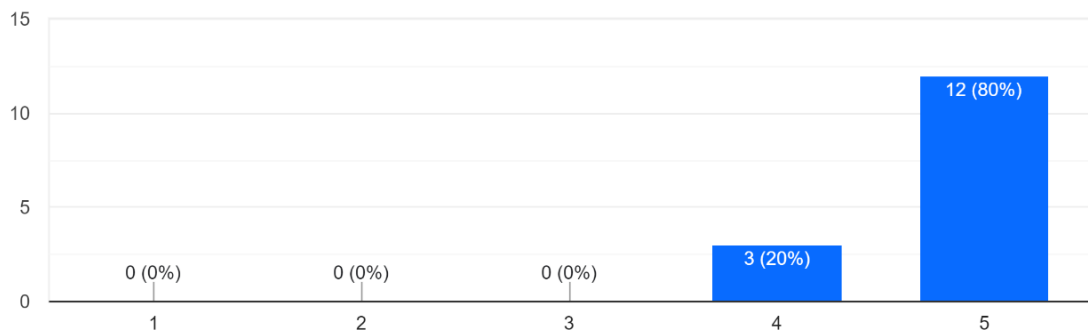


Fonte: Autoria própria.

Já no quesito do impacto sistema, as respostas também se demonstraram positivas (notas 4 e 5), conforme mostra a Figura 51.

Figura 51 - Gráfico da avaliação do grau do impacto do sistema

Avalie o grau de impacto que o protótipo causaria na área da saúde.
15 respostas



Fonte: Autoria própria.

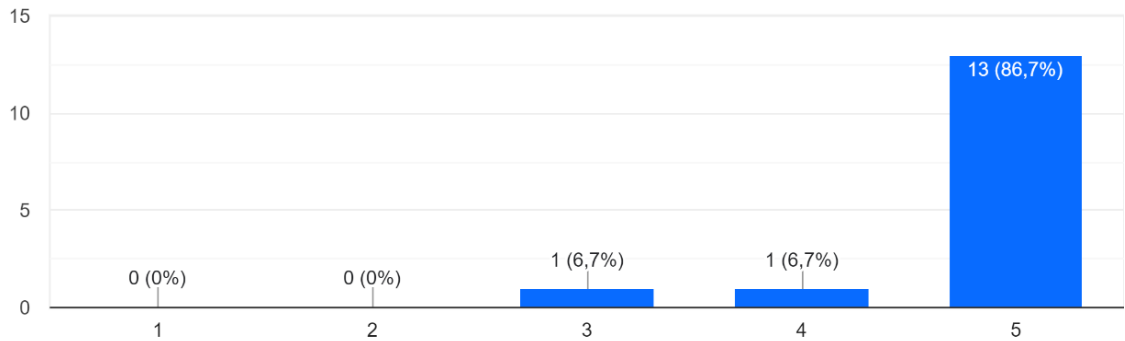
O gráfico da Figura 51 evidencia que, mesmo considerando a existência de sistemas de auto *check-in* em outros ambientes, essa funcionalidade seria uma novidade e traria benefícios significativos para a área de saúde.

Na avaliação da eficiência e eficácia do sistema de forma geral, foram obtidas algumas respostas positivas e uma resposta mediana (nota 3), conforme mostra a Figura 52.

Figura 52 - Gráfico da avaliação da eficiência e eficácia do sistema

Avalie a eficácia e eficiência do protótipo em resolver desafios ou lacunas identificadas na área da saúde.

15 respostas



Fonte: Autoria própria.

De acordo com a Figura 52, o sistema demonstra ser eficaz e eficiente em cumprir suas propostas, indicando possuir bom desempenho pelos usuários. No entanto, ao analisar a resposta mediana notada no gráfico, identificou-se um aspecto do sistema que não opera com total proveito. A análise do participante que deu nota 3 revelou pertencer ao enfermeiro, coincidindo com observações anteriores sobre o processo de encaminhamento descritas nas análises de fluxo do mesmo.

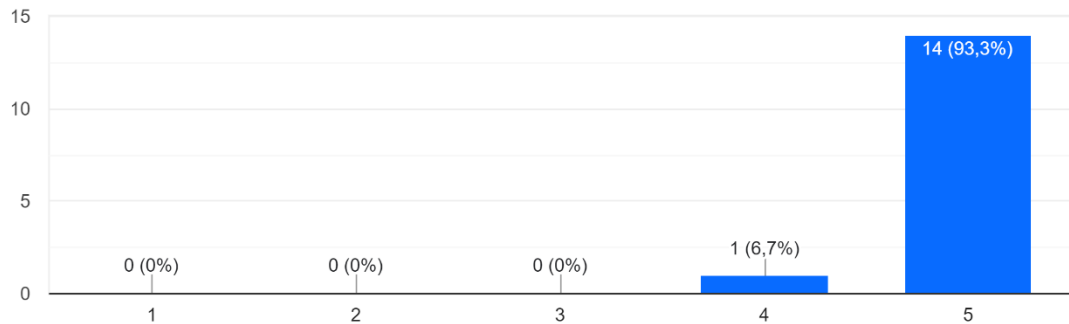
4.4.4 Satisfação e aceitação

Esta seção tem como objetivo avaliar o grau de satisfação e aceitação do sistema pelos usuários, aspectos que denotam a viabilidade da implementação deste sistema em um cenário real. Para tal avaliação, todos os entrevistados forneceram suas respostas analisando o sistema como um todo.

Os participantes avaliaram em notas 4 e 5 o grau de satisfação do sistema, um indicativo de que seu uso proporcionou uma experiência agradável, conforme exhibe o gráfico da Figura 53.

Figura 53 - Gráfico da avaliação da satisfação dos usuários ao utilizar o sistema

O quão satisfeito você está com a experiência geral proporcionada pelo protótipo?
15 respostas

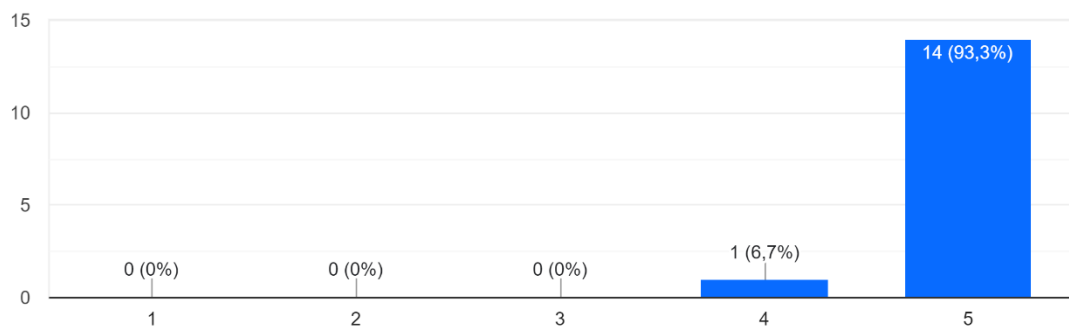


Fonte: Autoria própria.

Ademais, para avaliação da aceitação do sistema, obteve-se o mesmo resultado que o gráfico da Figura 53, com as notas em 4 e 5, demonstrando que a utilização deste sistema seria de grande valia para os usuários. Tal resultado é exibido na Figura 54.

Figura 54 - Gráfico da avaliação da aceitação do sistema

Avalie a possibilidade de um sistema deste tipo ser aceito na área da saúde.
15 respostas



Fonte: Autoria própria.

Os entrevistados justificaram suas respostas sobre a aceitação do sistema, as quais podem ser consultadas no apêndice K. Ao analisar tais repostas, é válido destacar que o sistema proporciona agilidade no atendimento de maneira eficiente, sendo esse o ponto crucial para tal aceitação. Além disso, outras justificativas se mostram como fatores favoráveis para aprimorar

esse aspecto, como a centralização de informações de forma simples, a redução das responsabilidades na recepção e a modernização do sistema de saúde.

Alguns comentários refletiram a perspectiva pós-Covid, pois a implementação deste sistema diminuiria o contato com pessoas em uma área altamente suscetível ao risco de contágio, sendo mais um ponto positivo. No entanto, houve apenas um comentário que indicou que a aceitação do sistema não seria totalmente favorável ao público idoso, o qual poderia enfrentar dificuldades em lidar com a tecnologia, mesmo assim, a avaliação correspondente, exibida no gráfico da Figura 54, foi boa (nota 4).

5 CONCLUSÃO

Os resultados obtidos a partir da implementação do sistema CheckInMed para inovação no setor médico indicam que ter um *software* dedicado a processos específicos, como agendamento de cirurgias e exames, pode aliviar significativamente as responsabilidades da recepção. O *check-in* automático nessas operações notifica a presença do paciente de forma eficiente, proporcionando visibilidade para todas as partes envolvidas e centralizando as informações, eliminando lacunas de comunicação.

As análises sobre a satisfação e aceitação dos usuários, apresentadas neste relatório, destacam a persistente necessidade, evidenciada pela pandemia de COVID-19, de implementar sistemas que agilizem o atendimento, reduzam o tempo de espera para os procedimentos médicos e minimizem o contato em áreas hospitalares. Logo, um sistema que incorpora tecnologias modernas, como o CheckInMed, pode tornar-se apto para implementação em um cenário hospitalar real.

Além do valor do *check-in* automático como aviso de presença para procedimentos, os resultados obtidos demonstram que as funcionalidades do *software* facilitam o trabalho dos profissionais de saúde, atendendo bem às suas necessidades gerais. No entanto, alguns pontos de melhorias foram observados, e um problema isolado foi identificado, que constitui na possibilidade de chamar vários pacientes com *check-in* realizado sobrecarregando a enfermaria. Tais considerações evidenciaram não só oportunidade de melhorar o sistema como também ressaltaram a importância da metodologia de aprimoramento contínuo para sistemas MVP.

Em suma, o sistema CheckInMed precisa que a automação do processo de encaminhamento de pacientes seja repensada para a viabilização do produto em um ambiente hospitalar. Contudo demonstrou ser bem-sucedido na integração de um sistema de gerenciamento de processos para a área de saúde, especialmente com foco no auto *check-in* para procedimentos médicos.

Trabalhos futuros

Considerando a escalabilidade do sistema MVP, ao se desenvolver uma nova versão para o sistema CheckInMed, é de grande valia a implementação de metodologias para deixá-lo mais robusto e profissional, sendo elas, balanceadores de cargas para distribuição de tráfego entre as múltiplas instâncias de servidores Node.js, *clustering* no Node.js para distribuição de carga de

trabalho nos processos do servidor, Redis para sistema de cache, *pools* de conexão no banco de dados para um tratamento mais eficiente do banco de dados gerenciando grandes volumes de cargas e reduzindo a sobrecarga, e microsserviços para maior flexibilidade das funções do sistema e sua manutenção.

Outro conceito que agrega valor e confiabilidade ao sistema são os testes automatizados, especialmente os testes E2E (*End-to-End*), que garantem o correto funcionamento dos fluxos do sistema. No *back-end*, utilizando Node.js, destaca-se o uso do *framework* Mocha juntamente com as bibliotecas Chai e Puppeteer. Já para o *front-end*, o *framework* Cypress é uma excelente escolha, garantindo testes robustos para as interfaces do usuário. Em relação à segurança, é recomendável o uso da ferramenta de teste Burp Suite, que avalia pontos de vulnerabilidade do sistema, assegurando sua proteção.

REFERÊNCIAS

- ANDRADE, F. *Dados sensíveis em uma requisição GET com HTTPS, por que não?*. [S. l.], 19 jul. 2019. Disponível em: <https://medium.com/@flaviochess/dados-sens%C3%ADveis-em-uma-requisi%C3%A7%C3%A3o-get-com-https-por-que-n%C3%A3o-87f1f9eea028>. Acesso em: 07 mai. 2024.
- BESSA, A. *Node.JS: o que é, como funciona esse ambiente de execução JavaScript e um Guia para iniciar*. 2023. Disponível em: <https://www.alura.com.br/artigos/node-js>. Acesso em: 25 abr. 2024.
- CASTRO, W. S.; LIMA, V. N.; COSTA, B. R. A. O novo modelo de gestão hospitalar pós-pandemia. 2020. *Trabalho Final (Curso Superior de Tecnologia em Gestão Hospitalar) - Faculdade Laboro*, [s. l.], 2020. Disponível em: <http://repositorio.laboro.edu.br:8080/jspui/handle/123456789/264?mode=full>. Acesso em: 09 mar. 2024.
- CHIMIELEWSKI, A. M. M.; SILVA, A. J. D. da; LEITE, J. R. E. Indústria 4.0: Revolução e impacto no mundo moderno. *Projectus*, v. 5, n. 1, p. 1–12, 2020.
- EWALLY. *Back-end: O Que É, Para Que Serve e Quais Suas Linguagens?*. 2023. Disponível em: <https://www.ewally.com.br/blog/ajudando-sua-empresa/back-end>. Acesso em: 15 mar. 2024.
- FIGMA. *Welcome to Figma Community*. 2024. Disponível em: <https://www.figma.com/community>. Acesso em: 09 mar. 2024.
- FREIRE, H. *React, Vue, Angular, conheça suas vantagens e desvantagens e qual é melhor para seus projetos*. 2019. Disponível em: <https://henrique-freire.medium.com/react-vue-angular-conhe%C3%A7a-suas-vantagens-e-desvantagens-e-qual-%C3%A9-melhor-para-seus-projetos-53734bb3d37f>. Acesso em: 07 mai. 2024.
- GARCIA, L. J.; BOTURA JUNIOR, G.; SILVA, J. C. R. P. da. Inovação e estratégia de marketing: uma revisão sistemática. *International Journal of Innovation*, São Paulo, v. 11, n. 3, p. 1-25, set./dez. 2023. Disponível em: <https://periodicos.uninove.br/innovation/article/view/23150>. Acesso em: 07 mai. 2024.

GRALHA, S. R. *et al.* Tecnologias de autoatendimento na área da saúde: um mapeamento sistemático. *Research, Society and Development*, v. 11, n. 9, p. e29511931926–e29511931926, 2022.

HOSTMÍDIA. *O que é MySQL e por que usar?*. 2024. Disponível em:

<https://www.hostmidia.com.br/blog/o-que-e-mysql/>. Acesso em: 14 mar. 2024.

IBERIA restablece la caída de hora y media de sus sistemas de check-in. Madrid: El Mundo, 26 abr. 2022. Disponível em:

<https://www.elmundo.es/economia/empresas/2022/04/26/6267bb10fdddffde2b8b4577.html>.

Acesso em: 18 abr. 2024.

LIMA, A. *Explorando as Vantagens e Desvantagens do Node.js*. 2023. Disponível em:

<https://devsagaz.com.br/explorando-as-vantagens-e-desvantagens-do-node-js/>. Acesso em: 06 mai. 2024.

LOUZADA, V.; CARVALHO, C.; LARANJA, E. *API: o que é, para quê serve e qual é a sua importância*. 2024. Disponível em: <https://www.alura.com.br/artigos/api>. Acesso em: 02 abr. 2024.

MARKET RESEARCH FUTURE. *Global Industrial Automation Market Overview*. 2023.

Disponível em: https://www.marketresearchfuture.com/reports/industrial-automation-market-2212?utm_term=&utm_campaign=&utm_source=adwords&utm_medium=ppc&hsa_acc=2893753364&hsa_cam=20823350561&hsa_grp=154923872134&hsa_ad=685356700925&hsa_src=g&hsa_tgt=dsa-2160750281009&hsa_kw=&hsa_mt=&hsa_net=adwords&hsa_ver=3&gad_source=1. Acesso em: 20 fev. 2024

MARQUES, R. *MariaDB 10: Ainda mais poderoso que o MySQL*. 2023. Disponível em:

<https://www.homehost.com.br/blog/tutoriais/mariadb-10-ainda-mais-poderoso-que-o-mysql/>.

Acesso em: 06 mai. 2024.

MATEUS, L. *Quem é melhor? ORM ou SQL puro?* 2023. Disponível em:

https://dev.to/lucas_jdev/quem-e-melhor-orm-ou-sql-puro-kjb. Acesso em: 07 mai. 2024.

MERCALDI, M.; BARCELLOS, E. I.; LANDIM, P. C.; BOTURA JUNIOR, G. Uma visão conceitual da inovação no design brasileiro. *In: INTERNATIONAL CONFERENCE ON*

INTEGRATION OF DESIGN, ENGINEERING AND MANEAGEMENT FOR INOVATION, 4., 2015, Florianópolis. *Anais [...]*. [S. l.: s. n.], 2015. Disponível em: <https://www.sorocaba.unesp.br/Home/Graduacao/EngenhariadeControleeAutomacao/galdenor/uma-visao-conceitual-da-inovacao-no-design-brasileiro.pdf>. Acesso em: 07 mai. 2024

MGITECH. Totens de autoatendimento no varejo. 2021. Disponível em: <https://www.mgitech.com.br/totem-de-autoatendimento#:~:text=Essencialmente%2C%20o%20totem%20%C3%A9%20um,sem%20intera%C3%A7%C3%A3o%20com%20agentes%20humanos>. Acesso em: 15 abr. 2024.

NASCIMENTO, J. *Introdução ao TypeScript*. 2023. Disponível em: https://dev.to/nascimento_/introducao-ao-typescript-3ekh#:~:text=Desvantagens%20do%20TypeScript%3A&text=Compila%C3%A7%C3%A3o%20adicional%20%3A%20Como%20o%20TypeScript,extra%20ao%20processo%20de%20desenvolvimento. Acesso em: 06 mai. 2024.

NEVES, V. *React: o que é, como funciona e um guia dessa popular ferramenta JS*. 2023. Disponível em: <https://www.alura.com.br/artigos/react-js>. Acesso em: 07 mai. 2024.

NEVES, V. *TypeScript: o que é, diferenças para o JS e como começar a aprender*. 2024. Disponível em: <https://www.alura.com.br/artigos/o-que-e-typescript>. Acesso em: 07 mai. 2024.

NOVO HAMBURGO. Prefeitura Municipal. *Protocolo por cores define ordem de atendimento dos pacientes*. Novo Hamburgo, 2020. Disponível em: <https://www.novohamburgo.rs.gov.br/noticia/protocolo-cores-define-ordem-atendimento-pacientes>. Acesso em: 01 mai. 2024.

ORACLE. *O que é o MySQL?*. 2024. Disponível em: <https://www.oracle.com/br/mysql/what-is-mysql/>. Acesso em: 25 abr. 2024.

PEREIRA, J. V.; SILVÉRIO, R. P. *Guia de JavaScript: o que é e como aprender a linguagem mais popular do mundo?*. 2023. Disponível em: <https://www.alura.com.br/artigos/javascript>. Acesso em: 25 abr. 2024.

RIES, E. *A startup enxuta*. [S.l.]: Leya, 2012.

SOUTO, M. *O que é Front-end Back-end e Full Stack - aprenda as diferenças entre essas áreas*. 2024. Disponível em: <https://www.alura.com.br/artigos/o-que-e-Front-end-e-back-end>. Acesso em: 14 mar. 2024.

VALENTIM, R. A. M. *et al.* Automação hospitalar: o estado da arte. *Revista Brasileira de Inovação Tecnológica em Saúde*, [Natal], v. 2, n. 1, p. 27-38, 2012. Disponível em: <https://periodicos.ufrn.br/reb/article/view/1929>. Acesso em: 09 mar. 2024.

VIEIRA, A. S. *et al.* O desenvolvimento tecnológico e as necessidades do consumidor geracional na hotelaria. *Turydes: revista sobre turismo y desarrollo local sostenible*, [s. l.], v. 10, n. 23, p. 12, 2017. Disponível em: <https://dialnet.unirioja.es/servlet/articulo?codigo=7911060>. Acesso em: 09 mar. 2024

VILLAIN, M.; SILVEIRA, M. I. *Figma: o que é a ferramenta, Design e uso*. 2023. Disponível em: <https://www.alura.com.br/artigos/figma#:~:text=Design%20%C3%A9%20uma%20%C3%A1rea%20que,de%20edi%C3%A7%C3%A3o%20ou%20apenas%20visualiza%C3%A7%C3%A3o>. Acesso em: 09 mar. 2024.

VYAS, K. *8 Major Advantages of Using MySQL*. 2023. Disponível em: <https://www.datamation.com/storage/8-major-advantages-of-using-mysql/>. Acesso em: 19 abr. 2024.

YANK SOLUTIONS. Com automação é possível ganhar até 80% na eficiência das tarefas operacionais, aponta Yank Solutions. [S. l.]: *GI*, 21 jun. 2023. Disponível em: <https://g1.globo.com/sp/bauru-marilia/especial-publicitario/yank-solutions/noticia/2023/06/21/com-automacao-e-possivel-ganhar-ate-80percent-na-eficiencia-das-tarefas-operacionais-aponta-yank-soghtml>. Acesso em: 20 fev. 2024.

GLOSSÁRIO

API (*Application Programming Interface*): Interface que permite a comunicação entre diferentes sistemas de *software*, facilitando a integração e troca de dados.

Back-end: Parte de um aplicativo ou site responsável pelo processamento de dados, lógica de negócios, e comunicação com o banco de dados e outras APIs.

Banco de dados: Sistema que armazena e organiza dados de maneira estruturada, permitindo consultas, atualizações e gerenciamento eficiente.

clearInterval: Função JavaScript que interrompe a execução repetida de um código ou função definida por *setInterval*.

clearTimeout: Função JavaScript que interrompe a execução agendada de um código ou função definida por *setTimeout*.

CSS (*Cascading Style Sheets*): Linguagem de estilo usada para definir a aparência e o *layout* das páginas *web*, incluindo cores, fontes, e posicionamento de elementos.

Deploy: Processo de colocar uma aplicação em produção, tornando-a acessível aos usuários finais.

DOM (*Document Object Model*): Representação em árvore dos elementos de uma página *web*, permitindo que scripts como JavaScript acessem e modifiquem a estrutura e o estilo do documento.

Endpoint: Ponto de entrada ou saída de uma API, representando uma URL específica onde uma determinada funcionalidade ou serviço pode ser acessado.

Figma: Ferramenta de design de interface baseada na *web*, permitindo colaboração em tempo real entre *designers* e desenvolvedores.

Frameworks: Conjuntos de ferramentas e bibliotecas que facilitam o desenvolvimento de *software*, fornecendo uma estrutura predefinida e funcionalidades reutilizáveis.

Front-end: Parte de um aplicativo ou site que interage diretamente com o usuário, incluindo a interface e a experiência do usuário.

Google Chrome: Navegador *web* desenvolvido pelo Google, conhecido por sua velocidade, segurança e suporte a padrões *web* modernos

Google Cloud Console: Interface *web* para gerenciamento dos serviços de nuvem oferecidos pelo Google Cloud, permitindo a administração de recursos e aplicativos.

Google Drive: Serviço de armazenamento em nuvem do Google que permite aos usuários armazenar, compartilhar e sincronizar arquivos online.

HTML (*HyperText Markup Language*): Linguagem de marcação usada para criar a estrutura e o conteúdo das páginas *web*.

HTTP (*Hypertext Transfer Protocol*): Protocolo de comunicação usado na transferência de dados na *web*, permitindo a troca de informações entre servidores e clientes.

JavaScript: Linguagem de programação usada principalmente para desenvolvimento *web*, permitindo a criação de páginas interativas e dinâmicas.

JSON (*JavaScript Object Notation*): Formato leve de intercâmbio de dados, fácil de ler e escrever para humanos, e fácil de analisar e gerar por máquinas.

Localhost: Endereço padrão usado para se referir ao computador local em uma rede, geralmente utilizado para desenvolvimento e testes.

MVP (*Minimum Viable Product*): Versão inicial de um produto com funcionalidades mínimas necessárias para validar a ideia com os usuários e obter *feedback*.

MySQL: Sistema de gerenciamento de banco de dados relacional de código aberto, amplamente utilizado para armazenamento e recuperação de dados.

Node.js: Ambiente de execução JavaScript do lado do servidor, permitindo a criação de aplicações de alta performance e escalabilidade.

ORM (*Object-Relational Mapping*): Técnica de mapeamento entre objetos de uma aplicação e tabelas de um banco de dados relacional, facilitando a manipulação de dados.

React.js: Biblioteca JavaScript desenvolvida pelo Facebook para a construção de interfaces de usuário, especialmente em aplicações de página única (SPA).

Render: Plataforma de *deploy* que facilita o processo de colocar uma aplicação em produção e torná-la acessível aos usuários finais.

sessionStorage: API *web* que permite armazenar dados no navegador de forma temporária, até que a sessão da página seja encerrada.

setInterval: Função JavaScript que executa um código ou função repetidamente em um intervalo de tempo especificado.

setTimeout: Função JavaScript que executa um código ou função uma única vez após um intervalo de tempo especificado.

SQL (*Structured Query Language*): Linguagem de consulta usada para gerenciar e manipular bancos de dados relacionais, permitindo operações como inserção, consulta e atualização de dados.

TCP/IP (*Transmission Control Protocol/Internet Protocol*): Conjunto de protocolos que permite a comunicação entre computadores em uma rede, sendo a base para a transmissão de dados na internet.

Templates: Estruturas predefinidas ou padrões usados para criar documentos ou componentes de *software* de forma rápida e consistente.

TypeORM: ORM específico para TypeScript e JavaScript, facilitando o uso de bancos de dados relacionais com essas linguagens.

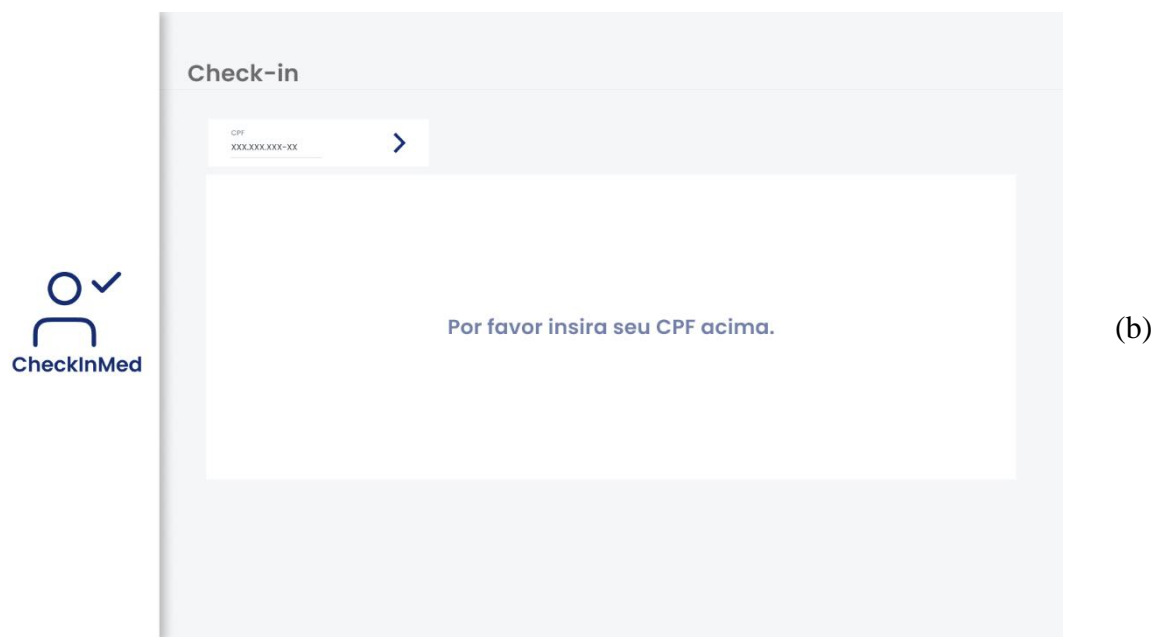
TypeScript: *Superset* do JavaScript que adiciona tipagem estática e outros recursos avançados, melhorando a qualidade do código e a manutenção.

URL (*Uniform Resource Locator*): Endereço utilizado para identificar e acessar recursos na internet.

Vercel: Plataforma de hospedagem de *Front-end* que facilita o *deploy* e a escalabilidade de aplicações *web*, especialmente as baseadas em JavaScript e React.

Web: Conjunto de informações e serviços acessíveis via internet, interconectados por URLs e visualizados através de navegadores.

APÊNDICE A – Design das telas planejadas para o sistema CheckInMed modeladas no software Figma





Check-in

CPF
XXXXXX-XX

Aviso!

Nenhum procedimento registrado para hoje. Por favor contate um responsável.

(c)



Check-in

CPF
XXXXXX-XX

Verifique seus dados

Informações Gerais

Nome Completo Fulano da Silva	Data de nascimento 02/05/2024	Nome da mãe Ciclano de Souza	E-mail fulaninho@email.com
CPF XXXXXX-XX	RG XX.XXX.XXX-X	Telefone (99)9999-9999	Telefone de emergência (99)9999-9999
Sexo Masculino	CRM SP-123456	COREN COREN-SP-123456	

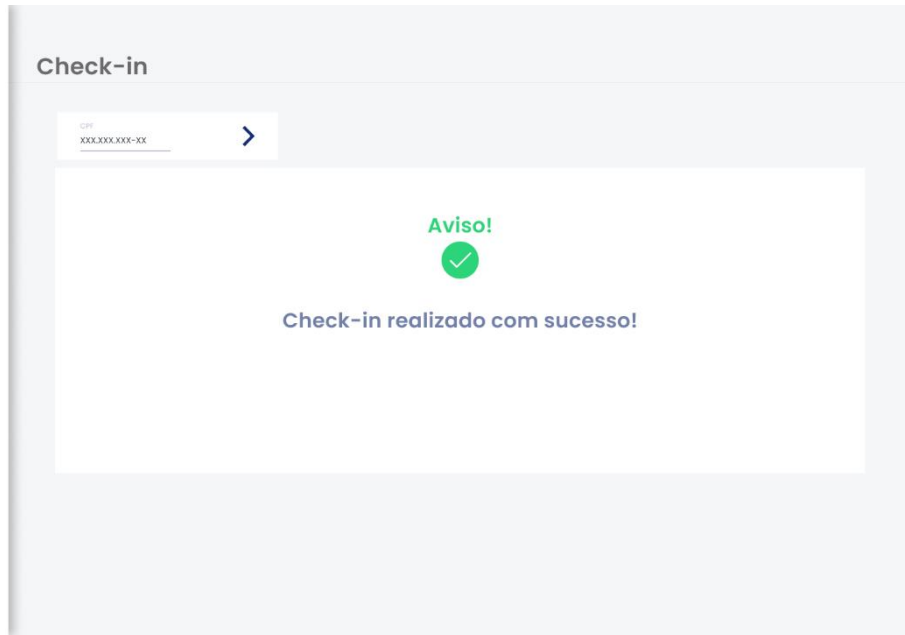
Endereço

Rua Rua dos paraísos	Número 1024	Complemento Apto 12	CEP 12.345-678	Bairro Jardim Bonito
Cidade São Paulo	Estado São Paulo			

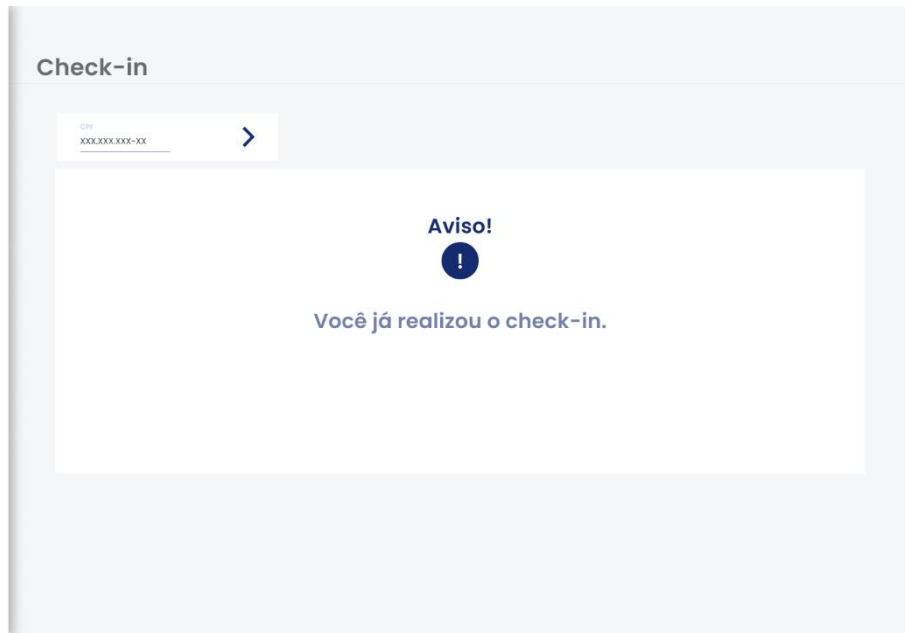
Procedimento

Nome do procedimento Desvio de septo	Médico responsável Dr. Manuel Fagundes	Data 12/12/2022	Hora 15:00
---	---	--------------------	---------------

(d)



(e)



(f)

CheckinMed

 Médico - Manuel
Manuel@email.com

- Usuários
- Procedimentos
- Perfil
- Encaminhamentos

Usuários

🔍 Pesquisar...

Nome	Tipo	Documento	Telefone	Ação
Lucas Silva	Paciente	000.000.000-00	(99)99999-9999	<input checked="" type="checkbox"/>
Patrícia Meideiros	Médico	000.000.000-00	(99)99999-9999	<input checked="" type="checkbox"/>
Hernán Lopes	Enfermeiro	000.000.000-00	(99)99999-9999	<input checked="" type="checkbox"/>

(g)

CheckinMed

 Médico - Manuel
Manuel@email.com

- Usuários
- Procedimentos
- Perfil
- Encaminhamentos

Informações de usuário

Informações Gerais

Nome Completo Fulano da Silva	Data de nascimento 02/05/2024	Nome da mãe Ciclano de Souza	E-mail fulaninho@email.com
CPF XXXXXXXXXX-XX	RG XX.XXXX.XXX-X	Telefone (99)99999-9999	Telefone de emergência (99)99999-9999
Sexo Masculino	CRM SP-123456	CREM DOREN-SP-123456	

Endereço

Rua Rua dos pardais	Número 1024	Complemento Apto 12	CEP 12.345-678	Bairro Jardim Bonito
Cidade São Paulo	Estado São Paulo			

(h)

CheckinMed

Médico - Manuel
Manuel@geral.com

- Uzários
- Procedimentos
- Perfil
- Encaminhamentos

Procedimentos

🔍 Pesquisar...

Paciente	Procedimento	Data	Médico responsável	Check-in	Hr. Check-in	Ação
● Lucas Silva	Desvio de septo	20/09/2024 16:00	Dr. Garcia Nunes	Presente	15:05:17	☑
● Patricia Meideiros	Lipospiração	07/07/2024 09:00	Dra. Fernanda Montenegro	Ausente	-	☑
● Hernán Lopes	Colonoscopia	03/04/2024 13:30	Dr. Luiz Guimarães	Presente	11:37:54	☑

(i)

CheckinMed

Médico - Manuel
Manuel@email.com

- 👤 Usuários
- 📋 Procedimentos
- 👤 Perfil
- 📅 Encaminhamentos

Detalhe do Procedimento

Informações Gerais

Paciente Fulano da Silva	Médico responsável Dr.º Pimpolho	Enfermeira Responsável Ciclana de Souza
Nome do procedimento Colonoscopia	Data 02/06/2024	Horário 00:00
	Horário do check-in 00:00	Check-in Presente

Exames

Exame 1	Exame 5
Exame 2	Exame 6
Exame 3	
Exame 4	+

Consultas

Motivo da consulta Quebra de dor de cabeça	Médico responsável Dr.º Pimpolho	Data da consulta 02/06/2024
Observações Paciente indica que as dores começaram ao comer um molho branco com um peixe.		
<hr/>		
Motivo da consulta Retorno	Médico responsável Dr.º Pimpolho	Data da consulta 22/05/2024
Observações Paciente fez uso de tratamento com medicamento e não houve melhora.		
<hr/>		

Histórico Médico

Alergias Camarão, maçã, leite
Histórico de doenças Catapora, caxumba, conjuntivite e pneumonia
Cirurgias anteriores Desvio de septo, transplante de córnea e uvulopalioplastia

Prontuário - Enfermeiro

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum eu auctor urna. Pellentesque finibus eros sit amet urna molestie mattis. Cras ac fringilla massa. Maecenas eu purus vitae lorem posuere sollicitudin. Duis elementum, nunc eget sollicitudin pharetra, nisi quam eleifend libero, eget pellentesque ligula ligula in risus. Curabitur a eros eu risus hendrerit tempor. Cras mauris purus, pretium id porttitor at, vehicula ut ante. Vestibulum maximus mi non lectus aliquam elementum. Fusce vestibulum mollis tellus ut scelerisque. Nulla vestibulum quam sed eros utricles, eget sagittis metus moncus. Aliquam turpis arcu, vehicula sed sagittis quis, interdum a nisi. Pellentesque sagittis pellentesque consectetur. Pellentesque id nulla sed nisi rutrum fringilla vitae tincidunt est. Vestibulum commodo vel sem id placerat. Fusce lectus metus, vehicula et justo sit amet, finibus efficitur sapien. Donec accumsan leo quis ex posuere cursus. Integer vel molestie neque. Nam sagittis pretium erat vel sodales. Quisque eget diam ligula. Quisque nec risus lobortis, imperdiet orci in, sollicitudin lectus. Nunc utricles justo est, sit amet tincidunt dui sodales cursus. Vivamus finibus ligula felis, quis sagittis lorem cursus quis. Vestibulum auctor urna sit amet tempor elementum.

Prontuário - Médico

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum eu auctor urna. Pellentesque finibus eros sit amet urna molestie mattis. Cras ac fringilla massa. Maecenas eu purus vitae lorem posuere sollicitudin. Duis elementum, nunc eget sollicitudin pharetra, nisi quam eleifend libero, eget pellentesque ligula ligula in risus. Curabitur a eros eu risus hendrerit tempor. Cras mauris purus, pretium id porttitor at, vehicula ut ante. Vestibulum maximus mi non lectus aliquam elementum. Fusce vestibulum mollis tellus ut scelerisque. Nulla vestibulum quam sed eros utricles, eget sagittis metus moncus. Aliquam turpis arcu, vehicula sed sagittis quis, interdum a nisi. Pellentesque sagittis pellentesque consectetur. Pellentesque id nulla sed nisi rutrum fringilla vitae tincidunt est. Vestibulum commodo vel sem id placerat. Fusce lectus metus, vehicula et justo sit amet, finibus efficitur sapien. Donec accumsan leo quis ex posuere cursus. Integer vel molestie neque. Nam sagittis pretium erat vel sodales. Quisque eget diam ligula. Quisque nec risus lobortis, imperdiet orci in, sollicitudin lectus. Nunc utricles justo est, sit amet tincidunt dui sodales cursus. Vivamus finibus ligula felis, quis sagittis lorem cursus quis. Vestibulum auctor urna sit amet tempor elementum.

Prescrição Médica

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum eu auctor urna. Pellentesque finibus eros sit amet urna molestie mattis. Cras ac fringilla massa. Maecenas eu purus vitae lorem posuere sollicitudin. Duis elementum, nunc eget sollicitudin pharetra, nisi quam eleifend libero, eget pellentesque ligula ligula in risus. Curabitur a eros eu risus hendrerit tempor. Cras mauris purus, pretium id porttitor at, vehicula ut ante. Vestibulum maximus mi non lectus aliquam elementum. Fusce vestibulum mollis tellus ut scelerisque. Nulla vestibulum quam sed eros utricles, eget sagittis metus moncus. Aliquam turpis arcu, vehicula sed sagittis quis, interdum a nisi. Pellentesque sagittis pellentesque consectetur. Pellentesque id nulla sed nisi rutrum fringilla vitae tincidunt est. Vestibulum commodo vel sem id placerat. Fusce lectus metus, vehicula et justo sit amet, finibus efficitur sapien. Donec accumsan leo quis ex posuere cursus. Integer vel molestie neque. Nam sagittis pretium erat vel sodales. Quisque eget diam ligula. Quisque nec risus lobortis, imperdiet orci in, sollicitudin lectus. Nunc utricles justo est, sit amet tincidunt dui sodales cursus. Vivamus finibus ligula felis, quis sagittis lorem cursus quis. Vestibulum auctor urna sit amet tempor elementum.


(j)


CheckinMed

 Médico - Manuel
Manuel@gmail.com




- Usuários
- Procedimentos
- Perfil
- Encaminhamentos

Encaminhamento

Início: 00:00 

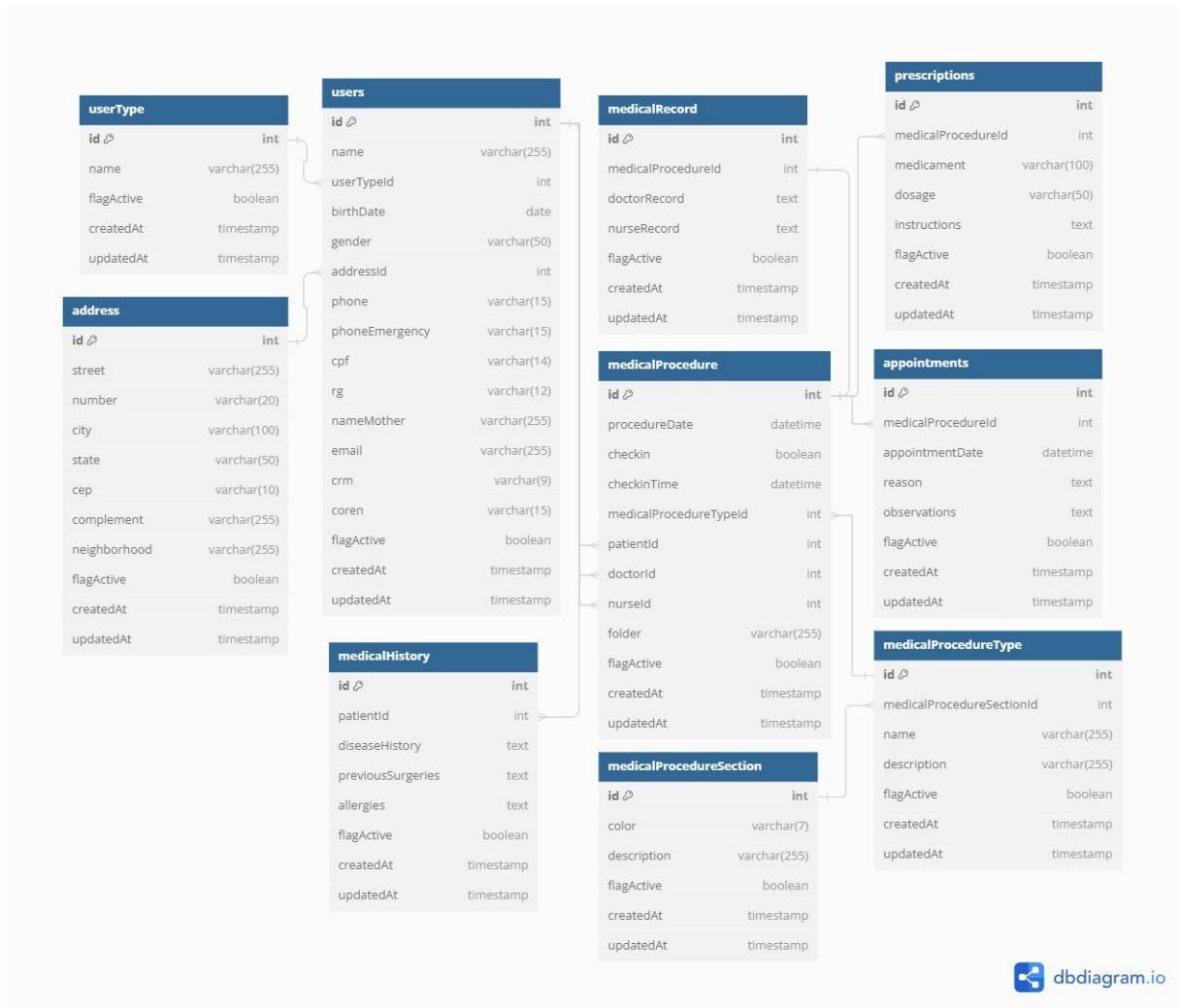
Fim: 00:00 

[Encaminhar](#)

Paciente	Procedimento	Data	Médico responsável	Check-in	Hr. Check-in	Ação
 Lucas Silva	Desvio de septo	20/09/2024 16:00	Dr. Garcia Nunes	Presente	15:05:17	<input checked="" type="checkbox"/>
 Patricia Meideiros	Lipospiração	07/07/2024 09:00	Dra. Fernanda Montenegro	Ausente	-	<input checked="" type="checkbox"/>
 Hernán Lopes	Colonoscopia	03/04/2024 13:30	Dr. Luiz Guimarães	Presente	11:37:54	<input checked="" type="checkbox"/>

(k)

APÊNDICE B – Modelo do banco de dados proposto



APÊNDICE C – Script de construção das tabelas do banco de dados MySQL gerado pelo software de modelagem dbdiagram.io para o sistema CheckInMed

```

CREATE TABLE `userType` (
  `id` int PRIMARY KEY AUTO_INCREMENT,
  `name` varchar(255),
  `flagActive` boolean,
  `createdAt` timestamp DEFAULT (current_timestamp()),
  `updatedAt` timestamp DEFAULT (current_timestamp())
);

CREATE TABLE `users` (
  `id` int PRIMARY KEY AUTO_INCREMENT,
  `name` varchar(255),
  `userId` int,
  `birthDate` date,
  `gender` varchar(50),
  `addressId` int,
  `phone` varchar(15),
  `phoneEmergency` varchar(15),
  `cpf` varchar(14),
  `rg` varchar(12),
  `nameMother` varchar(255),
  `email` varchar(255),
  `crm` varchar(9),
  `coren` varchar(15),
  `flagActive` boolean,
  `createdAt` timestamp DEFAULT (current_timestamp()),
  `updatedAt` timestamp DEFAULT (current_timestamp())
);

CREATE TABLE `address` (
  `id` int PRIMARY KEY AUTO_INCREMENT,
  `street` varchar(255),
  `number` varchar(20),
  `city` varchar(100),
  `state` varchar(50),
  `cep` varchar(10),

```

```

    `complement` varchar(255),
    `neighborhood` varchar(255),
    `flagActive` boolean,
    `createdAt` timestamp DEFAULT (current_timestamp()),
    `updatedAt` timestamp DEFAULT (current_timestamp())
);

CREATE TABLE `medicalProcedure` (
    `id` int PRIMARY KEY AUTO_INCREMENT,
    `procedureDate` datetime,
    `checkin` boolean DEFAULT (false),
    `checkinTime` datetime,
    `medicalProcedureTypeId` int,
    `patientId` int,
    `doctorId` int,
    `nurseId` int,
    `folder` varchar(255),
    `flagActive` boolean,
    `createdAt` timestamp DEFAULT (current_timestamp()),
    `updatedAt` timestamp DEFAULT (current_timestamp())
);

CREATE TABLE `medicalProcedureType` (
    `id` int PRIMARY KEY AUTO_INCREMENT,
    `medicalProcedureSectionId` int,
    `name` varchar(255),
    `description` varchar(255),
    `flagActive` boolean,
    `createdAt` timestamp DEFAULT (current_timestamp()),
    `updatedAt` timestamp DEFAULT (current_timestamp())
);

CREATE TABLE `medicalProcedureSection` (
    `id` int PRIMARY KEY AUTO_INCREMENT,
    `color` varchar(7),
    `description` varchar(255),
    `flagActive` boolean,
    `createdAt` timestamp DEFAULT (current_timestamp()),
    `updatedAt` timestamp DEFAULT (current_timestamp())
);

```

```
);  
CREATE TABLE `medicalHistory` (  
  `id` int PRIMARY KEY AUTO_INCREMENT,  
  `patientId` int,  
  `diseaseHistory` text,  
  `previousSurgeries` text,  
  `allergies` text,  
  `flagActive` boolean,  
  `createdAt` timestamp DEFAULT (current_timestamp()),  
  `updatedAt` timestamp DEFAULT (current_timestamp())  
);  
CREATE TABLE `appointments` (  
  `id` int PRIMARY KEY AUTO_INCREMENT,  
  `medicalProcedureId` int,  
  `appointmentDate` datetime,  
  `reason` text,  
  `observations` text,  
  `flagActive` boolean,  
  `createdAt` timestamp DEFAULT (current_timestamp()),  
  `updatedAt` timestamp DEFAULT (current_timestamp())  
);  
CREATE TABLE `prescriptions` (  
  `id` int PRIMARY KEY AUTO_INCREMENT,  
  `medicalProcedureId` int,  
  `medicament` varchar(100),  
  `dosage` varchar(50),  
  `instructions` text,  
  `flagActive` boolean,  
  `createdAt` timestamp DEFAULT (current_timestamp()),  
  `updatedAt` timestamp DEFAULT (current_timestamp())  
);  
CREATE TABLE `medicalRecord` (  
  `id` int PRIMARY KEY AUTO_INCREMENT,  
  `medicalProcedureId` int,  
  `doctorRecord` text,  
  `nurseRecord` text,
```

```
`flagActive` boolean,  
`createdAt` timestamp DEFAULT (current_timestamp()),  
`updatedAt` timestamp DEFAULT (current_timestamp())  
);  
  
ALTER TABLE `users` ADD FOREIGN KEY (`userId`) REFERENCES `userType`  
(`id`);  
  
ALTER TABLE `users` ADD FOREIGN KEY (`addressId`) REFERENCES `address`  
(`id`);  
  
ALTER TABLE `medicalProcedure` ADD FOREIGN KEY (`medicalProcedureTypeId`) REFERENCES `medicalProcedureType` (`id`);  
  
ALTER TABLE `medicalProcedure` ADD FOREIGN KEY (`patientId`) REFERENCES  
`users` (`id`);  
  
ALTER TABLE `medicalProcedure` ADD FOREIGN KEY (`doctorId`) REFERENCES  
`users` (`id`);  
  
ALTER TABLE `medicalProcedure` ADD FOREIGN KEY (`nurseId`) REFERENCES  
`users` (`id`);  
  
ALTER TABLE `medicalProcedureType` ADD FOREIGN KEY  
(`medicalProcedureSectionId`) REFERENCES `medicalProcedureSection` (`id`);  
  
ALTER TABLE `medicalHistory` ADD FOREIGN KEY (`patientId`) REFERENCES  
`users` (`id`);  
  
ALTER TABLE `appointments` ADD FOREIGN KEY (`medicalProcedureId`) REFERENCES  
`medicalProcedure` (`id`);  
  
ALTER TABLE `prescriptions` ADD FOREIGN KEY (`medicalProcedureId`) REFERENCES  
`medicalProcedure` (`id`);  
  
ALTER TABLE `medicalRecord` ADD FOREIGN KEY (`medicalProcedureId`) REFERENCES  
`medicalProcedure` (`id`);
```

APÊNDICE D – Script de população das tabelas do banco de dados MySQL para o sistema CheckInMed

```

INSERT INTO `userType` (`name`, `flagActive`, `createdAt`, `updatedAt`) VALUES
    ('Admin', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6)),
    ('Médico(a)', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6)),
    ('Enfermeiro(a)', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6)),
    ('Paciente', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6));

INSERT INTO `address` (`street`, `number`, `city`, `state`, `cep`, `complement`,
`neighborhood`, `flagActive`, `createdAt`, `updatedAt`) VALUES
    ('Rua João Dias de Souza', '365', 'Sorocaba', 'São Paulo', '18048-
090', 'Apto 11', 'Parque
Campolim', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6)),
    ('Rua Izabel Maria Rodrigues', '100', 'Sorocaba', 'São Paulo', '18078-
130', '', 'Jardim Santa
Cecília', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6)),
    ('Rua Fernando Rogich Vieira', '566', 'Sorocaba', 'São Paulo', '18071-
344', 'Casa 40', 'Jardim Santa
Helena', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6)),
    ('Rua Horácio Cenci', '400', 'Sorocaba', 'São Paulo', '18047-
800', '', 'Parque Campolim', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6)),
    ('Rua Irmã Dulce', '365', 'Sorocaba', 'São Paulo', '18079-685', 'Apto
45', 'Jardim Santa Madre
Paulina', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6)),
    ('Rua Doutor José Júlio Fernandes de Barros', '1001', 'Sorocaba', 'São
Paulo', '18046-100', 'Apto 72', 'Jardim dos
Estados', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6)),
    ('Rua Maria das Graças Arruda P. Nardy', '321', 'Sorocaba', 'São
Paulo', '18050-130', '', 'Jardim Vera
Cruz', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6)),
    ('Rua Quinze de Novembro', '40', 'Sorocaba', 'São Paulo', '18010-
080', 'Apto 33', 'Centro', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6)),
    ('Rua Renato Lucci', '123', 'Sorocaba', 'São Paulo', '18016-
329', '', 'Jardim Residencial
Martinez', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6)),
    ('Praça Maria Cristina Lolatta Pereira', '857', 'Sorocaba', 'São
Paulo', '18051-595', '', 'Jardim São
Paulo', 1, CURRENT_TIMESTAMP(6), CURRENT_TIMESTAMP(6));

INSERT INTO `users`
(`name`, `userId`, `birthDate`, `gender`, `addressId`, `phone`,
`phoneEmergency`, `cpf`, `rg`, `nameMother`, `email`, `crm`, `coren`, `flagActive`,
`createdAt`, `updatedAt`) VALUES
('Admin', 1, now(), '', null, '', '', '135.421.528-11', '', '', '', '',
1, now(), now()),

```

-- medics

```
('Allana Louise Olivia Gonçalves',2,'1988-01-17','Feminino',null,'(15) 99589-0415',
'', '027.004.288-11','32.704.923-6','Eliane Bárbara Simone','allana_goncalves@techdomus.com.br',
'SP-123456','',1,now(),now()),
('Daniel Filipe Luiz Oliveira',2,'1988-01-02','Masculino',null,'(15) 99374-5900',
'', '744.074.268-02','19.901.385-8','Sabrina Luiza','daniel_oliveira@mcexecutiva.com.br',
'SP-123456','',1,now(),now()),
('Anderson Victor Yago Farias',2,'1988-01-13','Masculino',null,'(15) 98104-2018',
'', '866.565.468-20','45.607.838-1','Sebastiana Marina','anderson_farias@tce.am.gov.br',
'SP-123456','',1,now(),now()),
```

-- nurses

```
('Isabelle Liz Gabrielly da Luz',3,'1968-02-03','Feminino',null,'(19) 3761-0239',
'', '019.853.288-10','42.194.512-6','Jaqueline Silvana Fátima','isabelle-daluz78@vinax.com.br',
'', 'COREN-SP-123456',1,now(),now()),
('Samuel Benedito Henry Cavalcanti',3,'1992-01-06','Masculino',null,'(11) 3795-4573',
'', '950.867.888-70','46.622.193-9','Clarice Bianca Adriana','samuel-cavalcanti71@jotace.eti.br',
'', 'COREN-SP-123456',1,now(),now()),
('Sarah Mariana Rocha',3,'2005-02-03','Feminino',null,'(19) 2918-6503',
'', '573.981.978-45','40.257.542-8','Rebeca Nina','sarahmarianarocha@agnet.com.br',
'', 'COREN-SP-123456',1,now(),now()),
('Rosângela Sara Jaqueline Galvão',3,'1993-02-24','Feminino',null,'(11) 3896-2825',
'', '046.480.068-44','41.894.123-3','Juliana Carolina','rosangela.sara.galvao@cabletech.com.br',
'', 'COREN-SP-123456',1,now(),now()),
('Marina Kamilly Melissa Lopes',3,'1979-02-24','Feminino',null,'(19) 3815-7046',
'', '530.657.328-23','36.162.967-9','Fernanda Esther','marina.kamilly.lopes@gmmail.com',
'', 'COREN-SP-123456',1,now(),now()),
```

-- patients

```
('Thales Benjamin da Costa',4,'2000-01-15','Masculino',1,'(15) 93953-5739',
'(15) 93953-5739','413.754.998-23','20.894.718-8','Allana Luiza Luana','thales_benjamin_dacosta@amplisat.com.br',
'', '',1,now(),now()),
('Aurora Jéssica Rosângela Fernandes',4,'1987-02-04','Feminino',2,'(15) 93872-5432',
'(15) 93872-5432','008.709.188-77','39.529.747-3','Jaqueline Alice Stefany','aurora_jessica_fernandes@santosferreira.adv.br',
'', '',1,now(),now()),
('Sara Yasmin Allana Cardoso',4,'1978-02-18','Feminino',3,'(15) 92733-3081',
'(15) 92733-3081','416.318.808-88','42.555.915-4','Louise Emily Tereza','sara_cardoso@nipbr.com',
'', '',1,now(),now()),
('Lorena Lorena Galvão',4,'1964-02-04','Feminino',4,'(15) 93757-6149',
'(15) 93757-6149','409.466.258-83','18.352.283-7','Cristiane Ayla Beatriz','lorenalorenagalvao@carubelli.com.br',
'', '',1,now(),now()),
```

```
('Rafael Breno Cavalcanti',4,'2006-01-11','Masculino',5,'(15) 92983-2122',
'(15) 92983-2122','093.272.158-34','18.907.298-2','Andreia
Hadassa','rafael-cavalcanti94@hpelzer.com.br','','',1,now(),now()),
```

```
('Isabel Eloá Ramos',4,'1951-01-08','Feminino',6,'(15) 93591-8594', '(15)
93591-8594','504.443.928-23','11.203.282-5','Vera
Bárbara','isabel_eloá_ramos@davimil.com.br','','',1,now(),now()),
```

```
('Sandra Sara da Paz',4,'1968-02-25','Feminino',7,'(15) 92979-1530', '(15)
92979-1530','320.808.308-48','48.428.829-5','Raimunda Sebastiana','sandra-
dapaz77@liv.com','','',1,now(),now()),
```

```
('Manoel Heitor Marcos Corte Real',4,'1975-01-06','Masculino',8,'(15)
93683-0707', '(15) 93683-0707','734.219.018-72','10.949.718-1','Agatha
Stefany Alice','manoel_cortereal@leandroreis.com','','',1,now(),now()),
```

```
('Renato Cláudio Caldeira',4,'1994-02-06','Masculino',9,'(15) 92687-7021',
'(15) 92687-7021','257.553.778-98','33.018.469-6','Alessandra
Tânia','renato_claudio_caldeira@embraer.com.br','','',1,now(),now()),
```

```
('Elza Andreia Andrea da Cruz',4,'1986-02-13','Feminino',10,'(15) 92804-
9300', '(15) 92804-9300','165.254.318-08','17.400.697-4','Yasmin Fabiana
Tatiane','elza.andreia.dacruz@htmail.com','','',1,now(),now());
```

```
INSERT INTO `medicalHistory`
```

```
(`patientId`,`diseaseHistory`,`previousSurgeries`,`allergies`,`flagActive`,
`createdAt`,`updatedAt`) VALUES
```

```
(10,'Infância e Adolescência: Infecções comuns, sem alergias
significativas.
```

```
Adolescência e Idade Adulta Jovem: Estilo de vida saudável, diagnóstico de
refluxo gastroesofágico (DRGE) em seus 20 anos.
```

```
Idade Adulta: Períodos de estresse, sem histórico de doenças
crônicas.','Apêndice removido aos 18 anos devido a apendicite aguda.
```

```
Cirurgia corretiva para hérnia umbilical aos 25 anos.','Alergias leves,
possivelmente sazonais.
```

```
Alergias a alimentos específicos, controladas com medicação quando
necessário.','1,now(),now()),
```

```
(11,'Histórico de asma desde a infância, controlada com medicação.
Diagnosticado com diabetes tipo 2 aos 40 anos.','Colecistectomia aos 50
anos devido a cálculos biliares.','Alergia a poeira e ácaros, sintomas
leves.',
```

```
1, now(), now()),
```

```
(12,'Sem histórico significativo de doenças na infância. Diagnosticado com
depressão aos 30 anos.','Cirurgia de apendicite aos 25 anos.','Alergia a
certos tipos de pólen, sintomas moderados.',
```

```
1, now(), now()),
```

```
(13,'Histórico de alergias sazonais desde a infância. Sofreu uma lesão no
joelho aos 28 anos, exigindo cirurgia de reconstrução do ligamento cruzado
anterior (LCA).','Nenhuma cirurgia anterior além da reconstrução do
LCA.','Alergia a certos tipos de frutos do mar, sintomas leves.',
```

```
1, now(), now()),
```

```

(14,'Diagnosticado com hipertensão aos 35 anos, controlada com medicação.
Cirurgia de herniorrafia umbilical aos 42 anos.','Cirurgia de herniorrafia
umbilical aos 42 anos.','Alergia a determinados medicamentos, sintomas
moderados.',

1, now(), now()),

(15,'Histórico de enxaqueca desde a adolescência, controlada com medicação.
Cirurgia de remoção de verruga aos 20 anos.','Cirurgia de remoção de
verruga aos 20 anos.','Alergia a determinados tipos de perfume, sintomas
leves.',

1, now(), now()),

(16,'Diagnosticado com artrite reumatoide aos 30 anos, tratamento contínuo
com medicamentos imunossupressores. Cirurgia de correção de desvio de septo
nasal aos 35 anos.','Cirurgia de correção de desvio de septo nasal aos 35
anos.','Alergia a pólen e poeira, sintomas moderados.',

1, now(), now()),

(17,'Sem histórico significativo de doenças na infância. Cirurgia de
correção de estrabismo aos 10 anos.','Cirurgia de correção de estrabismo
aos 10 anos.','Alergia a pelo de animais, sintomas leves.',

1, now(), now()),

(18,'Histórico de gastrite desde a adolescência, controlada com dieta e
medicação. Nenhuma cirurgia anterior.','Nenhuma cirurgia
anterior.','Alergia a certos tipos de frutas cítricas, sintomas leves.',

1, now(), now()),

(19,'Histórico de sinusite recorrente na infância e adolescência,
controlada com medicação. Nenhuma cirurgia anterior.','Nenhuma cirurgia
anterior.','Alergia a mofo e ácaros, sintomas leves.',

1, now(), now());

INSERT INTO medicalProcedureSection (color, description, flagActive) VALUES

  (#fc7703, 'Cirurgia', true), -- Laranja

  (#2ecc71, 'Exames', true), -- Verde

  (#ffdc00, 'Procedimentos', true); -- Amarelo

-- Inserir dados na tabela medicalProcedureType

INSERT INTO medicalProcedureType (medicalProcedureSectionId, name,
description, flagActive) VALUES

  (1, 'Cirurgia Cardíaca', 'Procedimento cirúrgico para tratar condições
cardíacas', true),

  (2, 'Ressonância Magnética', 'Exame de imagem para diagnosticar condições
internas', true),

  (3, 'Cirurgia de Emergência', 'Procedimento cirúrgico imediato em
situações críticas', true),

  (1, 'Cirurgia de Coluna', 'Procedimento cirúrgico para tratar condições
na coluna vertebral', true),

```

```

    (2, 'Tomografia Computadorizada', 'Exame de imagem que combina raios-X
para criar imagens detalhadas', true),

    (1, 'Cirurgia de Joelho', 'Procedimento cirúrgico para tratar condições
no joelho', true),

    (2, 'Ultrassonografia Abdominal', 'Exame de imagem para visualizar órgãos
internos no abdômen', true),

    (1, 'Cirurgia de Catarata', 'Procedimento cirúrgico para remover a
catarata no olho', true);

-- Inserir dados na tabela medicalProcedure

INSERT INTO medicalProcedure (procedureDate, checkin,
medicalProcedureTypeId, patientId, doctorId, nurseId, flagActive) VALUES

    ('2024-08-18 10:00:00', false, 1, 10, 2, 5, true),
    ('2024-07-19 14:30:00', false, 2, 10, 3, 6, true),
    ('2024-02-20 08:45:00', false, 1, 11, 2, 5, true),
    ('2024-09-21 11:20:00', false, 2, 11, 4, 6, true),
    ('2024-04-22 15:10:00', false, 1, 12, 3, 5, true),
    ('2024-02-23 09:30:00', false, 3, 13, 3, 6, true),
    ('2024-05-24 13:45:00', false, 3, 14, 3, 7, true),
    ('2024-03-25 10:30:00', false, 4, 15, 3, 8, true),
    ('2024-03-26 14:15:00', false, 5, 16, 4, 7, true),
    ('2024-03-27 09:00:00', false, 6, 17, 4, 8, true),
    ('2024-03-28 11:45:00', false, 4, 18, 4, 7, true),
    ('2024-02-29 15:30:00', false, 5, 19, 4, 8, true),
    ('2024-03-30 10:00:00', false, 7, 14, 2, 9, true),
    ('2024-03-31 14:45:00', false, 8, 16, 2, 9, true),
    ('2024-02-01 08:30:00', false, 7, 15, 2, 9, true);

-- Inserir dados na tabela appointments relacionados a cada
medicalProcedure

INSERT INTO appointments (medicalProcedureId, appointmentDate, reason,
observations, flagActive) VALUES

    (8, '2024-01-24 09:00:00', 'Consulta pré-cirúrgica', 'Avaliação antes da
cirurgia de coluna', true),

    (2, '2024-01-25 13:30:00', 'Avaliação de Imagem', 'Discussão dos
resultados da ressonância magnética', true),

    (4, '2024-01-25 13:30:00', 'Avaliação de Imagem', 'Discussão dos
resultados da ressonância magnética', true),

    (6, '2024-01-26 08:15:00', 'Consulta pré-cirúrgica', 'Avaliação antes da
cirurgia de joelho', true),

```

```

(13, '2024-01-27 10:45:00', 'Avaliação de Imagem', 'Discussão dos
resultados da ultrassonografia abdominal', true),

(15, '2024-01-27 10:45:00', 'Avaliação de Imagem', 'Discussão dos
resultados da ultrassonografia abdominal', true),

(14, '2024-01-28 14:30:00', 'Consulta pré-cirúrgica', 'Avaliação antes da
cirurgia de catarata', true),

(9, '2024-01-29 09:45:00', 'Avaliação de Imagem', 'Discussão dos
resultados da tomografia computadorizada', true),

(12, '2024-01-29 09:45:00', 'Avaliação de Imagem', 'Discussão dos
resultados da tomografia computadorizada', true),

(11, '2024-01-30 13:15:00', 'Consulta pré-cirúrgica', 'Avaliação antes da
cirurgia de coluna', true),

(1, '2024-01-31 08:45:00', 'Consulta pré-cirúrgica', 'Avaliação antes da
cirurgia de coração', true),

(3, '2024-01-31 08:45:00', 'Consulta pré-cirúrgica', 'Avaliação antes da
cirurgia de coração', true),

(5, '2024-01-31 08:45:00', 'Consulta pré-cirúrgica', 'Avaliação antes da
cirurgia de coração', true);

-- Inserir dados na tabela prescriptions

INSERT INTO prescriptions (medicalProcedureId, medicament, dosage,
instructions, flagActive) VALUES

(1, 'Paracetamol', '500mg', 'Tomar 1 comprimido a cada 6 horas', true),
(3, 'Ibuprofeno', '200mg', 'Tomar 1 comprimido a cada 8 horas', true),
(3, 'Amoxicilina', '500mg', 'Tomar 1 comprimido a cada 12 horas', true),
(1, 'Omeprazol', '20mg', 'Tomar 1 comprimido antes das refeições', true),
(5, 'Dipirona', '1000mg', 'Tomar 1 comprimido a cada 6 horas, conforme
necessário para dor', true),
(6, 'Ciprofloxacino', '500mg', 'Tomar 1 comprimido a cada 12 horas por 7
dias', true),
(8, 'Atenolol', '50mg', 'Tomar 1 comprimido por dia, de manhã', true),
(8, 'Loratadina', '10mg', 'Tomar 1 comprimido por dia, conforme
necessário para alergia', true),
(10, 'Paracetamol', '500mg', 'Tomar 1 comprimido a cada 6 horas', true),
(7, 'Paracetamol', '500mg', 'Tomar 1 comprimido a cada 6 horas', true),
(11, 'Paracetamol', '500mg', 'Tomar 1 comprimido a cada 6 horas', true),
(14, 'Omeprazol', '20mg', 'Tomar 1 comprimido antes das refeições',
true);

-- Inserir dados na tabela medicalRecord

INSERT INTO medicalRecord (medicalProcedureId, doctorRecord, nurseRecord,
flagActive) VALUES

```

(1, 'O paciente apresenta histórico de hipertensão. Recomendar acompanhamento regular da pressão arterial.', 'Paciente estável. Administrou medicamento conforme prescrição.', true),

(2, 'Resultados da ressonância magnética indicam inflamação na articulação do joelho esquerdo. Encaminhar para consulta com ortopedista.', 'Realizou avaliação de enfermagem. Recomendou repouso e aplicação de gelo.', true),

(13, 'Resultados da ultrassonografia abdominal indicam presença de pedra nos rins. Encaminhar para urologista.', 'Realizou avaliação de enfermagem. Recomendou aumento na ingestão de água e consulta com urologista.', true),

(5, 'Consulta pré-cirúrgica para cirurgia de catarata marcada para 2024-04-21. Informar paciente sobre procedimentos pré-operatórios.', 'Realizou avaliação pré-operatória. Explicou procedimentos e orientações pré-cirúrgicas.', true),

(6, 'Paciente submetido a cirurgia de emergência em 2024-02-22. Procedimento bem-sucedido. Recomendar acompanhamento pós-operatório.', 'Paciente em recuperação pós-operatória. Administrado medicamento conforme prescrição.', true),

(7, 'Paciente em recuperação pós-operatória após cirurgia de emergência. Avaliação médica regular recomendada.', 'Realizou avaliação de enfermagem pós-operatória. Monitorar sinais vitais.', true);

APÊNDICE E – Documento PDF que representa a prescrição médica para impressão, gerada para o navegador



CheckInMed

Receituário Médico

Paciente: Lorena Lorena Galvão

Gênero: Feminino

Data: 9/3/2024

Prescrição

1)

Medicamento: Ciprofloxacino

Dosagem: 500mg

Indicações: Tomar 1 comprimido a cada 12 horas por 7 dias

2)

Medicamento: teste

Dosagem: teste

Indicações: teste

Médico: Daniel Filipe Luiz Oliveira

CRM: SP-123456

Assinatura: _____

APÊNDICE F – Link do *deploy* do protótipo

Link de acesso ao protótipo: <https://check-in-med.vercel.app/>

QRCode de acesso ao protótipo:



APÊNDICE G – Seção de comentários de melhorias referente ao fluxo de Admin

Tem alguma sugestão do que o Fluxo de Admin poderia fazer ou deixar de fazer?

7 respostas

A opção de exames não ser habilitada para o administrador

Deixar somente a alteração de dados cadastrais do paciente para o fluxo de Admin, para não houver informação erroneamente no momento do check-in do paciente.

Médico e enfermagem somente eles colocarem exames do paciente.

Não

Sugestão de colocar opção de ordenação por coluna para ficar mais fácil de pesquisar nomes ou datas

No momento, não.

No momento não

No status do check-in, o display de ausente ou presente me remete muito a um botão, quando passo o mouse por cima do mesmo, descubro que não é um botão, talvez a troca do fundo vermelho por um ícone com a cor vermelha, ou verde para checkin efetuado, seria melhor

APÊNDICE H – Seção de comentários de melhorias referente ao fluxo de Médico

Tem alguma sugestão do que o Fluxo de Médico poderia fazer ou deixar de fazer?

9 respostas

Não

Sim, parte dos remédios de uso mais comum já poderiam aparecer como sugestão quando o médico fosse fazer a prescrição para o paciente

Não

Apenas criar a descrição da medicação, em campos específicos com colunas, para melhor visualização de todos no processo.

Mesma sugestão de layout sugerida para o Enfermeiro

No momento, não.

Apenas na parte de prontuário, onde ficaria de mais fácil funcionalidade para equipe médica os prontuários tanto medico como do enfermeiro ficassem em uma aba em comum, para melhor comunicação da equipe.

Sem sugestões

Algumas, uma delas é a receita, não precisa ter nome indicando o que são as coisas, exemplo, "medicamento", não precisa vir escrito, já se entende. Data, nome e CRM devem vir depois da assinatura. Outro ponto, é na prescrição médica tirar o X que divide o medicamento da dosagem, o traço funciona bem. As cores do procedimentos podem seguir as definidas pelo SUS de acordo com o grau de urgência (Vermelho, Amarelo, Verde e Azul).

APÊNDICE I – Seção de comentários de melhorias referente ao fluxo de Enfermeiro

Tem alguma sugestão do que o Fluxo de Enfermeiro poderia fazer ou deixar de fazer?

7 respostas

Não

Acho melhor apenas inserir o início e fim do atendimento pelo enfermeiro, sem restrição por faixa de horário

No encaminhamento deixar ele mais intuitivo para utilização, com horários. Assim como quando aparecer os pacientes colocar na frente do nome (lado esquerdo) um botão para o "encaminhamento" e depois dos minutos estipulados mudar para o botão "parar". Fica mais intuitivo e fácil para quando tiver mais de 1 paciente na tela. Regiões que não podem ser digitadas para "Detalhes do procedimento" podem ficar com uma cor diferente (um cinza escuro, por exemplo) e o que puder ser editado ficar na cor branca. Para a parte esquerda onde esta "procedimentos", "perfil" e "Encaminhamento" colocar o "Perfil" para região inferior dessa barra esquerda, e fazer isso para todos os usuários.

No momento, não.

O início e o fim na aba de encaminhamentos poderia ter um horário padrão como: Início = 00:00 e Fim = 23:59.

Na área do enfermeiro uma parte em que possa checar procedimentos médicos prescritos pelo médico e realizado pelo enfermeiro

APÊNDICE J – Seção de comentários de melhorias referente ao fluxo de Paciente

Tem alguma sugestão do que o Fluxo de Paciente poderia fazer ou deixar de fazer?

10 respostas

Não

Não, a abrangência das funcionalidades do Fluxo de Paciente são suficientes para a execução de todas as ações que podem ser necessárias para este tipo de usuário.

Colocar o tempo de duração do atendimento desde o momento do checkin

Não

Melhorar a indicação de como acessar e criar um indicador de campos a serem obrigatórios a preencher.

acredito que esteja bom o Fluxo de Paciente

No momento, não.

Deixar a área de exames para inclusão somente para equipe do hospital

Sem sugestões

Sim, ele só ter a opção de baixar seus exames, e apenas o médico poder incluir e excluir exames na tela de detalhe de procedimento. Ainda na tela de detalhe de procedimento, tirar tudo o que for relacionando a dado médico, deixar só informações do procedimento, a seção de "Informações gerais", pois deixando acesso a dados médicos pode ocorrer o caso de quebra de sigilo médico. A parte de check-in está muito boa.

APÊNDICE K – Justificativas de aceitação do sistema CheckInMed

Por favor, justifique o porquê da resposta acima.

15 respostas

Eficiência em integrar o sistema na saúde. Informações em um unico local

Sistema de fácil acesso e utilização, e onde medico é enfermagem tem uma grande importância de atualizar dados clínicos do paciente, onde fica fácil visualização para que todo o procedimento do paciente seja realizado de maneira correta e perfeita.

O gerenciamento eficiente na área da saúde é de suma importância não só para a otimização dos recursos disponíveis, mas também para salvar vidas, por isso, um sistema que promove este tipo de gerenciamento de forma fácil e intuitiva pode trazer melhorias significativas no modo como os hospitais brasileiros funcionam

O uso deste sistema possibilitaria um processo eficaz no atendimento de pacientes de um sistema de saúde

Considerando o fluxo em hospitais, é necessário q haja praticidade e funcionalidade nos departamentos.

Agilidade e registro em ambiente íntegro, onde todos envolvidos no processo, teriam como viabilizar e acompanhar o atendimento, minimizando desvios de informações, responsabilidades, tempo de espera e confiança nas informações do PACIENTE e seus acompanhantes, assim como os envolvidos em seus níveis de atendimento, passagem de turnos e organização de atendimento, leito, procedimentos, almoxarifado, remédios (controle), refeições, lavanderia, limpeza e segurança !!!

Mesmo sendo um aplicativo de grande utilidade podem gerar barreiras tecnológicas para pessoas de idade, e por se tratar da área da saúde existe a preocupação de consultórios estarem tendo práticas de etarismo.

Este produto ajuda a reduzir contágio em ambientes hospitalares, reduz carga de trabalho da recepção e facilita o tratamento de chamadas na área dos enfermeiros.

Pela praticidade e atendimento mais rápido dos pacientes.

Acredito que um sistema deste tipo engloba, dentro de uma área que, após o covid, está se digitalizando cada vez mais, as necessidades desse novo mundo digitalizado. Ter um sistema que te permite, com poucos cliques, realizar um check-in para a realização de consultas, prescrição de remédios e gerar receitas médicas, tudo online, é uma grande melhora na qualidade de vida

Quando nos encontramos em um sistema de saúde temos muitos problemas, seja de preenchimento de dados do paciente como o prontuário como a defasagem em sistemas para cirurgia. Com esse sistema facilitaria tanto a vida do paciente como da equipe médica, com eles a localidade do paciente, o prontuário, receituário ficariam localizado em um único sistema de fácil acesso para ambos, auxiliando na funcionalidade do dia a dia do hospital .

Pode agilizar o atendimento e o tempo de espera do paciente

Porque é um sistema que facilitaria bastante o dia a dia dos pacientes, médicos e enfermeiros, removendo a necessidade de um funcionário para fazer o check-in do paciente, otimizando o tempo deste processo e minimizando a probabilidade da ocorrência de erros durante a realização do mesmo.

O sistema iria ter um impacto no tempo de espera das pessoas ao chegarem no hospital, trazendo mais agilidade para o processo.

A utilidade do check-in ajudaria e muito em agilizar a comunicação e os processos já agendados no centro cirúrgico.