



UFOP

Universidade Federal
de Ouro Preto

**Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Departamento de Computação e Sistemas**

Um aplicativo para gestão do fluxo de equipamentos hospitalares móveis

Renato Júnior De Moura

TRABALHO DE CONCLUSÃO DE CURSO

ORIENTAÇÃO:

Gilda Aparecida de Assis

COORIENTAÇÃO:

Fernando Bernardes de Oliveira

**Novembro, 2022
João Monlevade–MG**

Renato Júnior De Moura

**Um aplicativo para gestão do fluxo de
equipamentos hospitalares móveis**

Orientador: Gilda Aparecida de Assis

Coorientador: Fernando Bernardes de Oliveira

Monografia apresentada ao curso de Sistemas de Informação do Instituto de Ciências Exatas e Aplicadas, da Universidade Federal de Ouro Preto, como requisito parcial para aprovação na Disciplina “Trabalho de Conclusão de Curso II”.

Universidade Federal de Ouro Preto

João Monlevade

Novembro de 2022



FOLHA DE APROVAÇÃO

Renato Júnior De Moura

Um aplicativo para gestão do fluxo de equipamentos hospitalares móveis

Monografia apresentada ao Curso de Sistemas de Informação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação

Aprovada em 04 de novembro de 2022

Membros da banca

Doutora - Gilda Aparecida de Assis - Orientadora (Universidade Federal de Ouro Preto)
Doutor - Fernando Bernardes de Oliveira - Coorientador - (Universidade Federal de Ouro Preto)
Doutora - Tatiana Alves Costa - (Universidade Federal de Ouro Preto)
Doutor - June Marques Fernandes - (Universidade Federal de Ouro Preto)

Gilda Aparecida de Assis, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 23/11/2022



Documento assinado eletronicamente por **Gilda Aparecida de Assis, PROFESSOR DE MAGISTERIO SUPERIOR**, em 23/11/2022, às 18:39, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0431108** e o código CRC **D78D7451**.

Esse trabalho é dedicado a minha mãe e minha avó, Mary e Luiza, e ao meu irmão, Eduardo, por fazerem parte dessa longa caminhada e por sempre acreditarem que eu seria capaz de chegar até aqui.

Agradecimentos

Agradeço primeiramente a, Deus por ter me acompanhado em mais um ciclo na minha vida, Foram anos de muita dedicação, esforços e aprendizado e que com certeza contribuíram não somente pela formação profissional mas como de vida

A minha mãe e avó, Mary e Luiza, por serem minha base, por todo incentivo. Ao meu irmão, Eduardo, por sempre acreditar em mim e me motivar a seguir adiante, buscando um futuro melhor.

A minha orientadora, Gilda Aparecida de Assis, por toda paciência, dedicação e motivação. Sem ela seria impossível a realização desse projeto.

A Universidade Federal de Ouro Preto (UFOP) por ter me oferecido toda estrutura necessária durante minha graduação. Além de todos os excepcionais profissionais que nela atuam e tive o prazer de compartilhar conhecimento e que contribuíram para o meu crescimento.

“Science is more than a body of knowledge; it is a way of thinking.”

— Carl Sagan (1934 – 1996),
in: The Demon-Haunted World: Science as a Candle in the Dark.

Resumo

Desde que foi detectada em dezembro de 2019, a COVID-19 se alastrou pelos diferentes continentes, tendo sido caracterizada como uma pandemia pela Organização Mundial da Saúde. A pandemia impactou, os sistemas de saúde que sofreram pressão decorrente da demanda extra gerada pela COVID-19. Nesse sentido, percebemos que o crescimento da pandemia aliado à precariedade dos sistemas de saúde para o atendimento de pacientes, culminaram em um grande fluxo nos serviços hospitalares, intensificando desperdícios. Um fluxo ineficiente desses processos pode comprometer o tratamento adequado de um paciente e, no pior cenário, ocasionar seu óbito. O aplicativo desenvolvido tem como objetivo auxiliar na redução de desperdícios relacionando o uso de equipamentos móveis nas emergências hospitalares como macas, suportes para soro e medicação. Esses desperdícios dizem respeito tanto a tempo de deslocamento para localizá-los em outros setores das unidades hospitalares ou mesmo nos espaços da emergência por onde eles circulam, e até mesmo registrar algum extravio e dessa forma reduzir também o tempo necessário, para buscar algum desses equipamentos no almoxarifado, localizando o mais próximo de sua localização atual através do aplicativo. O aplicativo não pretende substituir os sistemas de informação já existentes na unidade hospitalar, tendo como escopo apenas o fluxo dos equipamentos móveis de uso comum nas emergências hospitalares, buscando agilizar o processo de localização e histórico de uso desses equipamentos, sendo um aplicativo independente do controle de almoxarifado. como resultado foi obtido um aplicativo de gerenciamento do fluxo de equipamentos móveis para auxiliar na redução de desperdícios relacionados ao seu uso nas emergências. como também o reconhecimento e publicação do trabalho em eventos que buscam melhorias para a sociedade.

Palavras-chaves:Aplicação móvel, saúde, emergência hospitalar.

Abstract

Since it was detected in December 2019, COVID-19 has spread through the different continents, having been characterized as a pandemic by the World Organization for Health, The pandemic impacted, the health systems that suffered pressure from the extra demand generated by COVID-19. In this sense, we realize that with the great growth of the pandemic and the precariousness of health systems for the of patients on a large scale culminated in a large flow in hospital services intensifying waste. An inefficient flow of these processes can compromise the adequate treatment of a patient and in the worst scenario causing his death. The application to be developed is an application to help reduce waste by relating the use of equipment in hospital emergencies such as stretchers, IV stands and medication. These wastes concern both travel time to locate these equipment in other sectors of the hospital units. emergency where they circulate, and even register some loss and possibly reduce the time necessary to search for one of these elements in the warehouse, locating the closest to your location through the app. The app is not intended to replace the information systems that already exist in the hospital unit, having as scope only the flow of mobile equipment for common use in hospital emergencies, seeking to streamline the process of locating and using equipment history, being an application independent of warehouse control.as a result, a mobile equipment flow management application was obtained to help reduce waste related to its use in emergencies. as well as the recognition and publication of work in events that seek improvements for society.

Key-words: Mobile application, health, hospital emergency.

Lista de ilustrações

Figura 1 – <i>Diagrama de Casos de Uso</i>	33
Figura 2 – <i>Diagrama de Classe</i>	34
Figura 3 – <i>Protótipos: Splash Screen e Login</i>	35
Figura 4 – <i>Protótipos: Home, Pesquisa e Categoria</i>	36
Figura 5 – <i>Protótipos: Reserva</i>	37
Figura 6 – <i>Protótipos: Report</i>	38
Figura 7 – <i>Protótipos: Reserva e Devolução</i>	39
Figura 8 – <i>Hierarquia de pastas e arquivos da aplicação</i>	40
Figura 9 – <i>Hierarquia de pastas e arquivos da aplicação</i>	41
Figura 10 – <i>Arquivo JSON</i>	42
Figura 11 – <i>Conexão com API</i>	43
Figura 12 – <i>Conexão com API: Axios</i>	44
Figura 13 – <i>Conexão com API: Assíncrona</i>	44
Figura 14 – <i>Tela Inicial</i>	45
Figura 15 – <i>Tela De detalhes de Um Equipamento</i>	46
Figura 16 – <i>Tela de Reserva</i>	47
Figura 17 – <i>Tela de detalhes da reserva</i>	48

Lista de tabelas

Tabela 1 – Trabalhos Correlatos	24
Tabela 2 – Requisitos Funcionais	31
Tabela 3 – Requisitos Não Funcionais	32

Lista de abreviaturas e siglas

API Application Programming Interface

CRUD Create, Read, Update and Delete

IP Internet Protocol

NPM Node Package Manager

OTA Over the Air

QR Code Quick Response Code

SVG Scalable Vector Graphics

UML Unified Modeling Language

WEB World Wide Web

Sumário

1	INTRODUÇÃO	14
1.1	O problema de pesquisa	15
1.2	Objetivos	16
1.3	Organização do trabalho	17
2	REVISÃO BIBLIOGRÁFICA	18
2.1	Emergências Hospitalares	18
2.2	Equipamentos das Emergências	19
2.3	Trabalhos Correlatos	22
3	METODOLOGIA	25
3.1	Modelo em cascata	25
3.2	Tecnologias	26
3.2.1	React Native	27
3.2.2	Stack Navigator	27
3.2.3	JavaScript	27
3.2.4	Firebase	28
3.2.5	Visual Studio Code	28
3.2.6	Expo	29
4	PROJETO	30
4.1	Concepção do negócio	30
4.2	Requisitos Funcionais e Requisitos Não Funcionais	31
4.2.1	Requisitos Funcionais	31
4.2.2	Requisitos Não Funcionais	32
4.3	Diagramas UML	32
4.3.1	Diagrama De Caso De Uso	33
4.3.2	Diagrama De Classe	34
4.4	Protótipos	35
5	IMPLEMENTAÇÃO	40
5.1	Estrutura do Projeto	40
5.2	API	42
5.2.1	Conexão com o API	43
5.2.2	Entrada de Dados	43
5.2.3	Saída de Dados	43

5.3	Fluxo do aplicativo	44
6	CONCLUSÃO	49
6.1	Trabalhos Futuros	49
	REFERÊNCIAS	50

1 Introdução

A pandemia da Covid-19 iniciada em março de 2020 impôs diversas mudanças na vida das pessoas, incluindo a necessidade de distanciamento físico. Para os trabalhadores da saúde, setor considerado essencial, que estavam ou não no cuidado direto aos pacientes com Covid-19, estabeleceu-se um cenário único e sem precedentes. Eles estavam mais expostos ao risco de contaminação e morte, estavam submetidos a longas jornadas de trabalho, fadiga, exaustão (LANCMAN et al., 2021). Eles lidavam diariamente com o grande crescimento da pandemia e a precariedade dos sistemas de saúde para o atendimento de pacientes em larga escala, o que, conseqüentemente, culminou em um grande fluxo nos serviços hospitalares, intensificando problemas de falta ou uso inadequado de recursos.

No Brasil, a prestação de serviços na área da saúde caracteriza-se, de modo geral, pela cultura hospitalocêntrica. Constata-se que o hospital, mesmo diferenciando-se em tamanho, capacidade de atendimento e grau de complexidade, tem concentrado os atendimentos (SILVA et al., 2004; PIRES, 2008; PITTA, 1990)

O atendimento em saúde realiza-se por meio do trabalho humano que ocorre no exato momento em que é executado, como o que ocorre na produção do cuidado. A todo tempo esse trabalho requer o uso de instrumentos e máquinas, o cumprimento de normas, o relacionamento com a equipe e o contato com os pacientes (LANCMAN et al., 2021)

Diferentes fatores determinam a variabilidade do trabalho em saúde no contexto hospitalar, por exemplo, a diversidade das condições de saúde, os diferentes níveis de urgências e necessidades dos pacientes atendidos; os tempos heterogêneos de evolução da doença, de permanência no serviço e dos procedimentos realizados (exames, consultas, tratamentos); rotatividade de pacientes; quantidade e complexidade das informações tratadas e dos recursos utilizados. Além disso, o trabalho realiza-se em uma multiplicidade de locais por profissionais de diferentes áreas e competências, incluindo os setores assistencial, administrativo e apoio, com diversos tipos de vínculo e turnos de trabalho. Tais condições determinam diferentes responsabilidades e níveis hierárquicos na tomada de decisão e, por vezes, relações conflituosas entre setores e trabalhadores (LANCMAN et al., 2021).

O trabalho no contexto hospitalar é essencialmente coletivo e cooperativo. Diferentes pessoas executam atividades específicas que precisam ser coordenadas entre si. Ademais, os trabalhadores apoiam-se na realização de procedimentos em conjunto e compartilham informações visando realizar objetivos comuns. Os cursos das ações de cada trabalhador se entrelaçam, requerem sequência e continuidade, trocas de informações tanto na relação entre o usuário do serviço e o trabalhador, mas também entre os próprios trabalhadores em saúde (LANCMAN et al., 2021).

O riscos e cuidados a saúde são bastante relacionados , tendo em vista que risco compreende-se em situações , procedimentos que podem de certa forma serem negativos para o paciente, já situações de indisponibilidade de equipamentos nas emergências, seja por falta, mau funcionamento do mesmo, ou simplesmente por que ele não foi localizado em tempo hábil no local, representam situações de potencial risco para o paciente.

Neste contexto, o hospital tem sido objeto de estudos que envolvem: a) os pacientes, seus tratamentos e sua relação com a equipe de profissionais da saúde; b) as práticas médicas mais eficientes e eficazes; c) o controle de infecções; d) a organização hospitalar onde existem relações de saber-poder, saúde-doença, humanização da assistência, entre outras. O hospital também tem sido alvo de interesse enquanto espaço de trabalho, uma vez que, muitas vezes tem produzido condições potenciais de adoecimento de seus trabalhadores. tem despertado o interesse como empresa, impulsionando estudos sobre planejamento estratégico, sistemas de informação, redução de custos (WEBER; GRISCI, 2010).

Atualmente, os dispositivos móveis auxiliam as empresas a realizar o gerenciamento de seus processos, possibilitando maior vantagem competitiva no mercado. Assim, estes dispositivos podem acessar de forma instantânea as informações e podem ajudar a solucionar diversos tipos de problemas a qualquer hora, em qualquer lugar. De acordo com números o Brasil possui 242 milhões de celulares inteligentes, ao passo que a população brasileira, segundo último censo do Instituto Brasileiro de Estatística e Geografia (IBGE), é de de 210 milhões de habitantes, o que corresponde em média a mais de um celular para cada habitante (CNN, 2022). Estes dados mostram a disseminação dos smartphones que, cada vez mais, têm sido explorados por um amplo mercado de produtos e serviços. A área de saúde, mais especificamente o contexto hospitalar, pode se beneficiar do uso de smartphones em diferentes situações do trabalho, quer seja assistencial, administrativo ou de apoio.

Nesse contexto, o presente trabalho buscou elaborar uma solução por meio dos aplicativos móveis para melhorar o gerenciamento e controle do fluxo de equipamentos móveis nas emergências hospitalares. Espera-se que a implantação do aplicativo possa auxiliar a área de saúde, uma vez que a otimização de qualquer processo no setor de saúde, pode salvar vidas e facilitar a vida dos colaboradores que geralmente possuem rotinas longas e exaustivas.

1.1 O problema de pesquisa

Desde que foi detectada em dezembro de 2019, a COVID-19 vem se alastrando pelos diferentes continentes, tendo sido caracterizada como uma pandemia pela Organização Mundial da Saúde (OMS). Segundo a OMS, 80% dos pacientes com COVID-19 apresentam sintomas leves e sem complicações, 15% evoluem para hospitalização que necessita de oxigenoterapia e 5% precisam ser atendidos em unidade de terapia intensiva (UTI) (BULL

et al., 2020). Dependendo da velocidade de propagação do vírus na população, os sistemas de saúde podem sofrer forte pressão decorrente da demanda extra gerada pela COVID-19. Nesse sentido, percebemos que desde seu início, o Covid-19 agravou um sistema que já era falho, gerou consequências e desafios para a sociedade de caráter imediato, que tornaram urgente a gestão eficiente dos recursos, para fornecer o tratamento adequado e imediato à grande quantidade de pessoas que foram infectadas pelo vírus, como também às demais doenças que continuaram com seu percurso normal. O crescimento da pandemia, alinhado ao despreparo para o atendimento de pacientes, sobrecarregou os serviços hospitalares, o que consequentemente intensificou os desperdícios, tanto de materiais, insumos e equipamentos quanto de tempo dos profissionais de saúde. Um fluxo ineficiente desses processos pode comprometer o tratamento adequado de um paciente e, no pior cenário, ocasionar seu óbito. Tendo em vista que as emergências hospitalares têm sido muito afetadas por situações inesperadas como desastres, acidentes e pandemias como a Covid-19, elas carecem muitas vezes de agilidade no atendimento. Nesse sentido, o aplicativo proposto pretende auxiliar na redução dos tempos de deslocamento e de comunicação que os profissionais das emergências hospitalares gastam para localizar equipamentos móveis de uso comum nas emergências como macas, bombas de insulina, cadeiras de rodas, entre outros. O tempo de deslocamento para localizar e buscar os equipamentos pode envolver deslocamentos no próprio setor de emergência ou para outros setores do hospital. Muitos hospitais já adotam sistemas ERP (Enterprise Resource Planning) para a gestão de seus processos, como gestão de leitos, gestão de medicamentos, gestão de pessoas, gestão de enxovais, faturamento e gestão de equipamentos. Entretanto, muitos desses ERP hospitalares são baseados em desktops, e não exploram dispositivos móveis em toda sua potencialidade. O aplicativo proposto pretende utilizar os dispositivos móveis para registro e controle em tempo real do fluxo dos equipamentos móveis das emergências, ou seja, de onde o equipamento veio, para onde vai, quem fez a retirada, o equipamento ainda está em uso, etc. O aplicativo pode ser utilizado independentemente do processo já adotado pelo hospital para gestão de equipamentos.

1.2 Objetivos

O presente trabalho tem como objetivo principal desenvolver uma aplicação móvel que auxilie o gerenciamento de equipamentos móveis de uso comum no setor de emergência, promovendo uma redução de desperdícios. A aplicação permite registrar as trocas de equipamentos que ocorrem entre setores do hospital e profissionais durante as jornadas de trabalho. No intuito de facilitar a localização desses recursos nos espaços por onde eles circulam.

Os objetivos específicos são:

- Desenvolver um sistema responsivo a fim de possibilitar que o aplicativo possa se

adequar a diferentes dispositivos móveis;

- Desenvolver um aplicativo móvel multiplataforma para os sistemas operacionais Android e iOS a fim de alcançar um número maior de usuários.

1.3 Organização do trabalho

Este trabalho está estruturado da seguinte forma: O capítulo 1 apresenta a contextualização da proposta que será abordada neste estudo. No capítulo 2 são apresentados os trabalhos correlatos. No capítulo 3 é apresentada a metodologia de pesquisa percorrida por este trabalho, suas tecnologias e modelagem, Já no capítulo 4 é abordado o projeto do sistema proposto, incluindo os requisitos funcionais e não funcionais e diagramas UML. No capítulo 5 são apresentados os resultados obtidos e por fim no capítulo 6 é apresentado a conclusão do trabalho.

2 Revisão bibliográfica

2.1 Emergências Hospitalares

Entende-se por Serviços Hospitalares de Urgência e Emergência os chamados prontos-socorros hospitalares, pronto-atendimentos hospitalares, emergências hospitalares, emergências de especialidades, públicos e privados, civis e militares, excetuando-se os Serviços de Atenção às Urgências não Hospitalares, como as UPAs (Unidade de Pronto Atendimento) e similares.

Ao chegarem nas unidades hospitalares os pacientes são encaminhados para a triagem, onde são avaliados por um profissional, no intuito de identificar sinais e sintomas que indicam a urgência ou emergência no atendimento médico, para garantir a prioridade de atendimento aos pacientes com maior risco a vida, as unidades hospitalares adotam uma metodologia internacional, para a classificação de risco por meio do uso de pulseiras coloridas. Onde cada cor indica uma ordem de prioridade, durante a consulta o médico pode solicitar alguns exames como apoio ao diagnóstico, para então definir o tratamento adequado.

Mesmo que emergência e urgência sejam termos distintos e que tenham significados diferentes na medicina, geralmente são usadas como sinônimas. Ambos os termos promovem atendimento médico rápido e proporcional à gravidade. Entretanto, a emergência exige um tratamento direto por conta do risco iminente de morte ou lesão permanente, como fraturas expostas, paradas cardiorespiratórias e hemorragias graves. Já a urgência, apesar de demandar uma pronta avaliação médica por seu risco potencial, não necessariamente precisa de uma intervenção instantânea, como nas fraturas não expostas, cólicas renais, aumento de pressão arterial, entre outras. O atendimento de urgência e emergência ocorre quando o paciente necessita de uma ação médica imediata (MOTTA; MENA; PIACSEK, 2017).

Define-se por urgência a ocorrência imprevista de agravo à saúde com ou sem risco de vida, cujo portador necessita de assistência médica imediata, ou seja, as situações de urgência são aquelas que não apresentam risco de morte iminente, mas que devem ser tratadas rapidamente para não evoluir para complicações mais graves. Em geral, as principais situações que precisam de um atendimento de urgência são:

- Torções, luxações e fraturas;
- Transtornos psiquiátricos;
- Dor abdominal de intensidade moderada;

- Febre maior que 38 graus há mais de 48 horas;
- Retenção urinária em pacientes idosos.

Define-se por emergência, a constatação médica de condições de agravamento à saúde que levam a um risco iminente de vida ou sofrimento intenso, necessitando, portanto, de tratamento médico imediato. Na emergência é necessário ter o diagnóstico e iniciar o tratamento o mais rápido possível para que os riscos sejam minimizados e a pessoa esteja em segurança. Algumas das principais situações que necessitam de atendimento imediato, pois existe um risco potencial de morte, são:

- Acidente de origem elétrica;
- Picada ou mordida de animais peçonhentos;
- Cortes e queimaduras profundas;
- Hemorragia;
- Infarto;
- Derrame;
- Sangue na urina, no vômito, nas fezes ou ao tossir;
- Reação alérgica grave;
- Alta dificuldade respiratória;
- Convulsões;
- Febres altas e constantes;
- Acidentes de trânsito (incluindo atropelamentos)

2.2 Equipamentos das Emergências

Os equipamentos são os itens imprescindíveis para o diagnóstico, manutenção e tratamento dos pacientes. Eles são reutilizáveis, sendo desinfetados ou esterilizados para novo uso. Já os insumos são de uso único, descartáveis e eliminados logo após o procedimento. Assim, os equipamentos médicos hospitalares se referem aos aparelhos e ao mobiliário da unidade de saúde (MORSCH, 2018).

Os equipamentos de diagnóstico são ferramentas ou aparelhos empregados com o objetivo de diagnosticar as condições de saúde dos pacientes. A partir dos sintomas descritos pelo indivíduo e avaliados pelo médico, esse tipo de recurso é empregado para investigar

o paciente mais profundamente, buscando pelas anomalias no organismo possivelmente responsáveis pelos quadros observados.

Já os equipamentos de tratamento são aqueles voltados à terapia de condições específicas, ou seja, eles são utilizados para restaurar a função dos tecidos ou órgãos afetados por certos quadros patológicos, ou ainda para dar auxílio a eventuais intervenções cirúrgicas.

Os equipamentos de suporte à vida são aqueles utilizados quando o organismo do paciente é incapaz de desempenhar suas funções por conta própria. Assim, como o próprio nome diz, são aparelhos voltados à manutenção da função corporal dos indivíduos comprometidos por algum quadro adverso de saúde.

Os equipamentos médicos hospitalares são duráveis, sendo empregados para uso contínuo contra determinadas doenças ou condições. Seu uso é de longo prazo, alguns podem ser empregados tanto no hospital quanto na própria casa do paciente, e visam fornecer todo o suporte, conforto e segurança necessários durante o tratamento de certos quadros mais complexos.

Segundo(MORSCH, 2018), durante a pandemia Covid-19, alguns equipamentos foram utilizados em larga escala, são eles:

- Ventiladores pulmonares , que administram o fluxo de ar e controlam a troca de gases nos pulmões dos pacientes, a fim de mantê-los vivos durante quadros de insuficiência respiratória, que são típicos da COVID-19 grave;
- Bombas de infusão, que gerenciam automaticamente os fluidos ministrados nos pacientes para nutrição ou infusão de medicamentos, o que é imprescindível para programar e controlar os sedativos no CTI (Centro de Tratamento Intensivo), por exemplo;
- Eletrocardiógrafo, que permitem verificar a velocidade e o volume de batimentos cardíacos, sendo fundamentais para monitorar pacientes que têm o sistema circulatório e cardíaco comprometidos pelo agravamento da Covid;
- Monitor multiparamétrico, que é obrigatório para acompanhar os sinais vitais dos pacientes internados no CTI e também pode ser empregado nos setores de emergência, de recuperação, triagem, cirúrgico, entre outros.

Em um levantamento baseado na literatura e na observação informal da dinâmica em um hospital regional, foram identificados os seguintes equipamentos móveis de uso comum nas emergências:

- **Desfibrilador;** equipamento que tem como função descarregar cargas elétricas na parede torácica ou nas fibras musculares do coração
- **Oxímetro;** equipamento que determina fotoeletricamente a saturação de oxigênio.
- **Estetoscópio;** equipamento utilizado para escutar sons vasculares, respiratórios, cardíacos e o trato digestório
- **Eletrocardiógrafo;** equipamento com o qual se realiza o eletrocardiograma – teste que monitora a atividade elétrica do coração.
- **Maca hospitalar;** são as camas de salas de emergência. Sua principal característica é a mobilidade por terem rodinhas e serem dobráveis.
- **Hamper;** utilizado para transportar roupas sujas, infectadas e contaminadas sem ter contato com outros ambientes.
- **Mesa auxiliar;** utilizada para guardar os utensílios e equipamentos médicos de forma organizada facilitando o transporte destes.
- **Equipamento de raio-x;** equipamento que serve para realizar exames de radiografia, que são como fotografias da parte interna do corpo. Por meio dessas imagens, é possível observar estruturas anatômicas, como ossos, órgãos e vasos sanguíneos, sem precisar de cortes ou incisões.
- **Carrinho de emergência;** tem como principal objetivo a facilitação do acesso à equipe médica e de enfermagem aos medicamentos, equipamentos e materiais de emergência de forma rápida e dinâmica.
- **Cadeira para coleta de sangue;** é utilizada para coleta de sangue, aplicação de injeção, aferimento de pressão arterial.
- **Cardioversor;** equipamento que aplica um choque elétrico sobre o tórax com o objetivo de despolarizar as fibras cardíacas de maneira simultânea. Ele visa restaurar o impulso do coração de forma coordenada.
- **Aparelho de pressão;** equipamento utilizado para aferir a pressão sanguínea de maneira rápida e descomplicada.
- **Balança antropométrica;** equipamento eletrônico de precisão, utilizado em hospitais, clínicas, farmácias e consultórios médicos para pesagem em unidades kg e gramas (g).
- **Cama hospitalar;** equipamento utilizado para o tratamento de enfermos em hospitais, postos de saúde ou internação domiciliar.

- **Autoclave;** equipamento que possibilita tratamento térmico bastante utilizado no ambiente hospitalar e que consiste em manter o material contaminado a uma temperatura elevada, através do contato com vapor de água, durante um período de tempo suficiente para destruir todos os agentes patogênicos.
- **Bomba de insulina;** permite um melhor controle da glicemia, pois oferece um fluxo de insulina constante, em doses precisas, de forma que não é necessário utilizar as injeções de insulina, sendo também uma opção mais confortável para o tratamento do diabetes.
- **Suporte de soro;** equipamento que auxilia no manuseamento e no transporte do soro quando utilizado no paciente.

Ainda no contexto de equipamentos móveis de uso comum nas emergências, foram feitas pesquisas na literatura e na observação informal, para entender as características do fluxo de gerenciamento de equipamentos hospitalares, foram identificados as seguintes etapas.

- **Inventário;** fazer um inventário e descrever a quantidade exata dos equipamentos médico hospitalares, sendo indispensável para manutenção dos equipamentos.
- **Classificação;** classificar os equipamentos e dividi-los em grupos.
- **Gerenciar os espaços físicos;** Todo espaço ocupado por esses equipamentos está relacionado ao gerenciamento de localização, isso vai desde as salas que você escolhe para alocá-los, as ferramentas utilizadas para isso, os equipamentos e a manutenção em si.
- **Tecnologia;** investir em tecnologias para um sistema de codificação de equipamentos.
- **Treinamento;** capacitar os colaboradores, para que estejam aptos a utilizar as tecnologias que irão fazer o gerenciamento dos equipamentos hospitalares.

2.3 Trabalhos Correlatos

No contexto de aplicativos móveis para gerenciamentos de equipamentos hospitalares foi realizada uma pesquisa nas lojas de aplicativos Play Store e App Store no mês de maio de 2022 e foram encontrados os seguintes aplicativos que são relevantes ao escopo do projeto:

HCleitos: Aplicativo para gerenciamento de leitos hospitalares. Trata-se de um sistema utilizado no hospital do coração de Dourados,MS, que contém funções para o

gerenciamento de qualquer equipamento hospitalar como manutenção e solicitação de agendamentos.

Hospitalar: Aplicativo que permite cadastrar colaboradores do hospital e gerenciar escala de plantões.

Também foi realizada uma pesquisa no Google acadêmico com os termos de busca "aplicação móvel" e "emergência" e foram selecionados os seguintes trabalhos:

- **Aplicando código QR e aplicativo móvel para melhorar o processo de atendimento no hospital tailandês**

O artigo fala sobre a superlotação hospitalar que tem sido um problema no sistema de saúde público tailandês, e como principal causa a precariedade e escassez de recursos, como médicos, enfermeiros. Neste trabalho, um sistema de baixo custo foi desenvolvido e testado em um hospital público com orçamento limitado. O sistema utilizou código QR e aplicativo de smartphone para capturar os processos hospitalares e o tempo gasto em atividades individuais. Com os dados de atividade disponíveis, dois algoritmos foram desenvolvidos para identificar duas grandezas que são valiosas para conduzir a melhoria do processo: o tempo mais congestionado e as atividades gargalo. O sistema foi implantado em um hospital público e os resultados foram apresentados (CHAROENSIRIWATH et al., 2015).

- **Projeto e implementação de sistema de gerenciamento de equipamentos móveis baseado em QRcode**

O trabalho apresenta um sistema de gerenciamento de equipamentos móveis baseado em QRcode é proposto para gerenciamento remoto e conveniente de dispositivos. Ao contrário dos sistemas convencionais, o sistema aqui torna os gerentes acessíveis a informações em tempo real com smartphones. Comparado com o método convencional, que só pode ser operado com dispositivos específicos, este modo de telegestão leve e eficiente é propício para a gestão de ativos em vários cenários (YU; DUAN; JIAO, 2017).

- **Sistema de informação de gestão de equipamentos médicos usando dispositivos móveis**

Este estudo, aborda sobre como a grande quantidade de tecnologias está diretamente incorporada aos dispositivos móveis e como eles podem ajudar no gerenciamento das informações, o estudo teve como objetivo o desenvolvimento de um sistema de informação móvel para gerenciamento de equipamentos médicos. Aonde a plataforma central do sistema possibilita a ligação do dispositivo móvel a um servidor web, para tratar das informações referentes aos equipamentos médicos. através da leitura de códigos QR-Code, que possibilitam identificar os equipamentos médicos em um

banco de dados, e as informações relevantes sobre o equipamento médico em questão. Este Projeto em seu estado atual é uma ferramenta básica de apoio à manutenção de equipamentos médicos. É também uma alternativa moderna, competitiva e econômica no mercado atual (NÚÑEZ; CASTRO, 2011).

A [Tabela 1](#) Compara os trabalhos correlatos selecionados.

Tabela 1 – Trabalhos Correlatos.

	QrCode	Funcionamento	Utilização	Armazenamento
HCleitos	Não	Online/Offline	Multiplataforma	Sim
Hospitalar	Não	Online/Offline	Dispositivo Movel	Sim
Charoensiriwath	Sim	Online/Offline	Multiplataforma	Sim
Yu,Duan,Jiao	Sim	Online/Offline	Multiplataforma	Sim
Núñez,Castro	Sim	Online	Multiplataforma	Sim

Fonte: Elaborado pelo autor (2022)

3 Metodologia

A pesquisa aqui apresentada é uma pesquisa exploratória, que busca identificar objetos de estudo dentro do contexto de aplicações móveis em emergências hospitalares, além da problematização do fluxo de equipamentos nas emergências hospitalares. Quanto à natureza, a pesquisa é aplicada, uma vez que tem como objetivo “gerar conhecimentos para aplicação prática, dirigidos à solução de problemas específicos” (GERHARDT, 2009).

Os procedimentos metodológicos da pesquisa são:

- **Fundamentação Teórica:** Pesquisa bibliográfica sobre o funcionamento das emergências hospitalares, assim como dos equipamentos que são comumente utilizados durante as rotinas do ciclo comum dos serviços hospitalares.
- **Estudo de trabalhos correlatos:** foi realizado um levantamento na Google Play Store e na App Store para conhecimento das soluções que já estão disponíveis no mercado, assim como publicações a fim de proporcionar entendimento das suas características e escopo.
- **Seleção de ferramentas:** Análise das plataformas, sistemas operacionais, linguagens e banco de dados a serem utilizados.
- **Modelagem:** Levantamento dos requisitos funcionais e não funcionais, modelagem do sistema seguindo o modelo em cascata e construção de diagramas.
- **Implementação:** Desenvolvimento do software utilizando a linguagem e bibliotecas selecionados anteriormente.
- **Testes:** Execução de testes unitários e de integração e validações no sistema desenvolvido.

Nas próximas seções são apresentados o modelo de desenvolvimento adotado e as ferramentas selecionadas para o estudo.

3.1 Modelo em cascata

O modelo em cascata foi criado em 1970 e representa um modelo de desenvolvimento de software sequencial, no qual o seguimento de uma atividade para a outra depende exclusivamente da finalização da atividade anterior.

Para Sommerville (2011), o modelo em cascata deve ser usado apenas quando os requisitos são bem compreendidos e é improvável que venham a ser alterados durante o

desenvolvimento do sistema. O modelo em cascata reflete o tipo de processo usado em outros projetos da engenharia. Como é mais fácil utilizar um modelo de gerenciamento comum para todo o projeto, processos de software baseados no modelo em cascata ainda são comumente utilizados.

Segundo [Pressman \(2006\)](#) O modelo cascata descreve um método de desenvolvimento linear e seqüencial. Na primeira vez que uma fase de desenvolvimento é completada, o desenvolvimento prossegue para a próxima fase e não há retorno. A vantagem do desenvolvimento cascata é que ele permite controle departamental e gerencial. Um planejamento pode ser atribuído com prazo final para cada fase de desenvolvimento e um produto pode prosseguir no processo de desenvolvimento e, teoricamente, ser entregue no prazo. O desenvolvimento parte do conceito, através do projeto (design), implementação, teste, instalação, descoberta de defeitos e termina com a operação e manutenção. Cada fase de desenvolvimento segue em uma ordem estrita, sem qualquer sobreposição ou passos iterativos.

3.2 Tecnologias

A priori, para o desenvolvimento de uma aplicação móvel online, foi necessária uma busca das principais tecnologias disponíveis no mercado. Foi realizado um levantamento das plataformas, sistemas operacionais e banco de dados que para desenvolver a proposta.

Uma solução geral que considera desenvolver, manter, testar e implantar soluções computacionais para diferentes plataformas é um tópico importante. ([WU, 2018](#))

Um dos objetivos do projeto é criar uma aplicação móvel que seja disponível tanto para sistema operacional Android quanto para o iOS, pois, de acordo com uma pesquisa realizada pela ([GARTNER, 2017](#)), Android e iOS representam 99,6% de todas as vendas de smartphones até o quarto trimestre de 2016.

Nesse sentido, atualmente há duas alternativas de desenvolvimento disponíveis no mercado de aplicações móveis, sendo elas, desenvolvimento nativo e desenvolvimento multiplataforma.

- **Plataforma Nativa:** São aplicações desenvolvidas com base no código particular de cada sistema operacional, utilizando ferramentas e linguagens de programação específicas. As aplicações nativas possuem, naturalmente, melhor desempenho e fácil acesso aos componentes do smartphone, pois são compilados em código de máquina otimizado para o sistema operacional em uso ([MATOS; BRITTO, 2016](#)).
- **Multiplataforma:** São aplicações desenvolvidas a partir de um único código-fonte, que podem ser executadas em diferentes sistemas operacionais. As aplicações mul-

tiplataformas têm maior facilidade no desenvolvimento e manutenção (MATOS; BRITTO, 2016).

No presente projeto optou-se por uma abordagem de desenvolvimento multiplataforma, com os aplicativos renderizados nativamente para os sistemas operacionais Android e iOS.

3.2.1 React Native

React Native é uma extensão do React, também chamado de ReactJS ou React.js, que foi lançado em 2015 e projetado para desenvolvimento de software móvel nativo. React Native utiliza a linguagem JavaScript, uma linguagem de programação que visa permitir o uso de funcionalidades móveis nativas e de integração (PETE, 2017).

O React Native é utilizado por organizações em todo o mundo para criar aplicativos de telefone/tablet multiplataforma a partir de um código base. Isso significa é possível escrever código facilmente que será executado em iPhones, iPads, telefones Android e tablets sem ter que reescrevê-lo. Além disso, esses aplicativos são executados nativamente e podem ser implantados na Apple AppStore ou na Google Play Store. Portanto, eles são mais rápidos e confiáveis(NATIVE, 2020).

Uma das principais características do React Native é que ele apresenta técnicas modernas da web para dispositivos móveis, sem comprometer os recursos e o desempenho. Os aplicativos do React Native são escritos principalmente em JavaScript e operam no núcleo do JavaScript(WU, 2018).

3.2.2 Stack Navigator

O Stack Navigator fornece uma maneira do aplicativo realizar a transição entre as telas, sendo que cada nova tela é colocada no topo de uma pilha. Por padrão, o navegador de pilha é configurado para ter a aparência familiar do iOS e do Android, ou seja, novas telas deslizam da direita no iOS e utilizam a animação padrão do Android.

3.2.3 JavaScript

JavaScript é uma linguagem de programação de alto nível, interpretada e estruturada, com tipagem dinâmica fraca e multiparadigma, criada para ser usada em navegadores e páginas web, sendo considerada atualmente uma das principais tecnologias da Web.

Segundo (MOZILLA, 2022), JavaScript é uma linguagem de programação leve, orientada a objetos, baseada em protótipos e em funções de primeira classe, conhecida como a linguagem de script da Internet.

Mas o JavaScript não se restringe apenas às páginas e aos navegadores. Com o advento de diversos frameworks, APIs, melhorias e criação de centenas de funções, hoje já é possível utilizar JavaScript em aplicativos móveis, softwares para desktop e até mesmo em aplicações do tipo back-end.

De acordo com uma pesquisa do([OVERFLOW, 2021](#)) o JavaScript é a linguagem de programação mais implantada, sendo eleita por nove anos consecutivos como a linguagem de programação mais utilizada.

3.2.4 Firebase

O Firebase é uma plataforma que inclui um banco de dados na nuvem, plataforma que tem sido largamente utilizada para aplicações web. Ele armazena os dados em Formato JavaScript Object Notation (JSON)([KHAWAS; SHAH, 2018](#)).

O Firebase Realtime Database é um banco de dados NoSQL, baseado em nuvem, que sincroniza dados em todos os clientes em tempo real e oferece funcionalidade offline. Os dados são armazenados no banco de dados Realtime como JSON e todos os clientes conectados compartilham uma instância, recebendo automaticamente atualizações com os dados mais recentes([MORONEY, 2017](#)). Firebase facilita a transferência de arquivos de forma fácil e segura, independentemente da rede, sendo considerado um serviço de armazenamento de objetos econômico. O desenvolvedor pode usá-lo para armazenar imagens, áudio, vídeo ou outros conteúdo gerados pelos usuários([KHAWAS; SHAH, 2018](#)).

Realtime Database fornece uma linguagem de regras declarativas que permitem definir como os dados devem ser estruturados, indexados e também quando podem ser lidos e gravados. Por padrão, o acesso de leitura e gravação ao banco de dados é permitido para usuários autenticados.

Uma API é fornecida ao desenvolvedor do aplicativo para que os dados do aplicativo sejam sincronizados entre clientes e armazenados na nuvem do Firebase. As bibliotecas cliente são fornecidas pelo próprio Firebase, que permite a integração com Aplicativos Android, IOS e JavaScript ([KHAWAS; SHAH, 2018](#)).

3.2.5 Visual Studio Code

O Visual Studio Code é um editor de texto multiplataforma, disponibilizado pela Microsoft para o desenvolvimento de aplicações web, com suporte principalmente a ASP.NET e Node.js ([DEVMEDIA, 2016](#)).

Optou-se pela utilização da IDE Visual Studio Code por apresentar uma interface amigável, ser uma ferramenta gratuita e muito simples, mas que pode ser customizada com diversas extensões, ser multiplataforma e possuir recursos de depuração e análise de código fonte, o que conseqüentemente permite que os usuários possam contribuir com melhorias.

3.2.6 Expo

Expo é uma ferramenta de desenvolvedor para criar experiências com gestos e gráficos interativos, usando JavaScript e React Native. Ela apresenta muitos recursos para criar e dimensionar um aplicativo, como atualizações Over the Air ([OTA](#)), compartilhamento instantâneo do aplicativo e suporte na web.

Expo é uma estrutura de software usada em conexão com o React Native para facilitar o desenvolvimento. A Expo oferece várias ferramentas diferentes, como bibliotecas e caminhos de desenvolvimento, para facilitar e possibilitar o desenvolvimento de diferentes maneiras. Os caminhos de desenvolvimento contidos no Expo são recursos que permitem ao desenvolvedor implementar o aplicativo independentemente do sistema operacional. Além disso, as bibliotecas oferecidas pela Expo são parte da operação desses caminhos de desenvolvimento. Em termos práticos, uma aplicação React Native é criada usando o framework de software da Expo e ambos os caminhos de desenvolvimento e seu uso são ilustrados ([TENHUNEN, 2021](#)).

4 Projeto

Um projeto de software é uma descrição da estrutura do software a ser implementado, dos modelos e estruturas de dados usados pelo sistema, das interfaces entre os componentes do sistema e, às vezes, dos algoritmos usados. Os projetistas não chegam a um projeto final imediatamente, mas o desenvolvem de forma iterativa. Eles acrescentam formalidade e detalhes enquanto desenvolvem seu projeto por meio de revisões constantes para correção de projetos anteriores (PRESSMAN; MAXIM, 2021).

A priori, o projeto consiste na concepção do negócio, ou seja, a gestão do fluxo dos equipamentos móveis das emergências hospitalares realizada por colaboradores que atuam durante os plantões.

4.1 Concepção do negócio

A etapa de concepção no desenvolvimento de software tem como objetivo auxiliar no desenvolvimento da ideia de produto que o cliente deseja, a partir da compreensão de qual problema o produto se propõe a solucionar, quem serão os seus usuários e como ele irá gerar valor para o negócio (PEDROZO, 2018).

Na concepção do negócio são considerados aspectos sobre o entendimento do problema e as possíveis soluções e por fim a prototipação para efetivar e testar as ideias de solução.

Nesse sentido, são realizadas as seguintes etapas para a concepção do negócio:

- **Imersão:** Entendimento do problema.
- **Definição:** Definir o problema a ser resolvido.
- **Ideação:** Etapa que contribui para materializar a ideia do produto, pois tem o objetivo de decidir o que irá ser desenvolvido e porque irá ser desenvolvido.
- **Estruturação:** Etapa mais técnica, utilizada para tomar decisões de quais tecnologia serão utilizadas no produto, seus requisitos e peculiaridades.
- **Prototipagem:** Etapa para testar as ideias de solução.

A concepção do negócio foi elaborada a partir de observações de situações vivenciadas nas emergências em um hospital local, quando o autor do trabalho exercia o papel de paciente ou de acompanhante. Durante esses períodos, o autor observou situações de

comunicação face a face entre a equipe de enfermagem na busca de equipamentos, como bomba de insulina e macas.

4.2 Requisitos Funcionais e Requisitos Não Funcionais

Os requisitos de um sistema são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições a seu funcionamento. Esses requisitos refletem as necessidades dos clientes para um sistema que serve a uma finalidade, como controlar um dispositivo, colocar um pedido ou encontrar informações. O processo de descobrir, analisar, documentar e verificar esses serviços e restrições é chamado engenharia de requisitos (PRESSMAN; MAXIM, 2021).

4.2.1 Requisitos Funcionais

Requisitos funcionais são declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a entradas específicas e de como o sistema deve se comportar em determinadas situações. Em alguns casos, os requisitos funcionais também podem explicitar o que o sistema não deve fazer.

A [Tabela 2](#) descreve os requisitos funcionais do sistema proposto.

Tabela 2 – Requisitos Funcionais.

Código	Requisito Funcional	Prioridade
RF001	O software deve permitir que o usuário faça sua autenticação	Alta
RF002	O software deve permitir que o usuário visualize todos os equipamentos por categoria	Alta
RF003	O software deve permitir que o usuário reserve, reagende e devolva equipamentos	Alta
RF004	O software deve permitir que o usuário faça a leitura de equipamentos por QR Code	Alta
RF005	O software deve permitir que o usuário, visualize todos os equipamentos por localidade	Alta
RF006	O software deve conter fotos dos equipamentos, local de origem e para local de destino e a identificação de quem levou	Média
RF007	O software deve permitir que o usuário reporte um equipamento	Baixa

Fonte: Elaborado pelo autor (2022)

4.2.2 Requisitos Não Funcionais

São restrições aos serviços ou funções oferecidos pelo sistema. Incluem restrições de timing, restrições no processo de desenvolvimento e restrições impostas pelas normas e legislação vigentes. Ao contrário das características individuais ou serviços do sistema, os requisitos não funcionais, muitas vezes, aplicam-se ao sistema como um todo.

De acordo com [Pressman e Maxim \(2021\)](#), os requisitos não funcionais, como o nome sugere, são requisitos que não estão diretamente relacionados aos serviços específicos oferecidos pelo sistema a seus usuários. Eles podem estar relacionados às propriedades emergentes do sistema, como confiabilidade, tempo de resposta e ocupação de área.

A [Tabela 3](#), descreve os requisitos não funcionais do sistema proposto.

Tabela 3 – Requisitos Não Funcionais.

Código	Requisito Funcional	Prioridade
RF001	O software terá tempo de carregamento dos dados menor que três segundos	Média
RNF002	O usuário necessitará de no máximo 20 segundos para carregar a tela atual do software	Baixa
RNF003	O software deverá ser acessível para pessoas com idade avançada	Baixa
RNF004	O usuário necessitará de no máximo cinco interações com o software para finalizar uma tarefa	Média
RNF005	O software deverá funcionar de maneira online	Alta
RNF006	O software deve possuir versões para Android e IOS	Média
RNF007	O software deverá consumir menos de 20MB de memória	Baixa

Fonte: Elaborado pelo autor (2022)

4.3 Diagramas UML

A UML(Unified Modeling Language) ou Linguagem de Modelagem Unificada é uma linguagem visual utilizada para modelar softwares baseados no paradigma de orientação a objetos. É uma linguagem de modelagem de propósito geral, que pode ser aplicada a todos os domínios da aplicação. Essa linguagem tornou-se, nos últimos anos, a linguagem-padrão de modelagem adotada internacionalmente pela engenharia de software([GUEDES, 2009](#)).

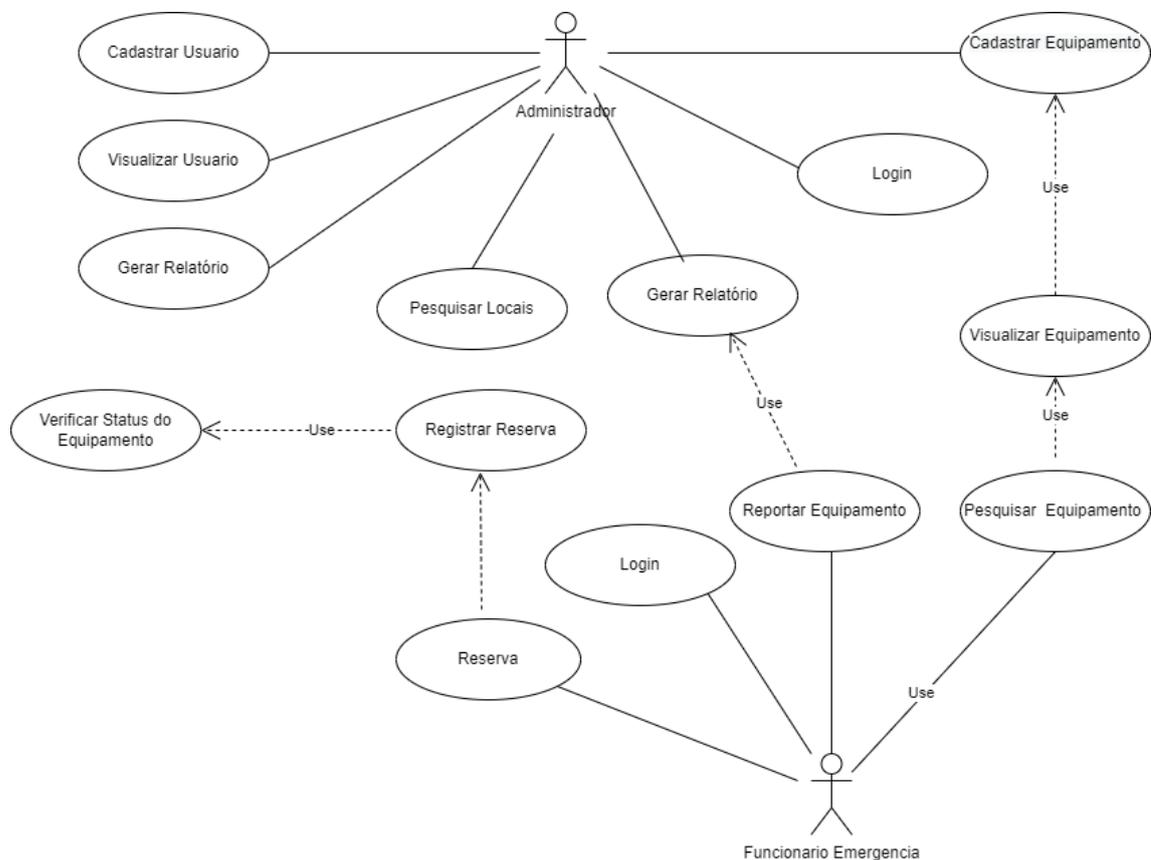
Foram selecionados diagramas para auxiliar na elaboração e desenvolvimento do sistema. A utilização dos diagramas Unified Modeling Language ([UML](#)) neste trabalho tem por objetivo modelar as características e funcionalidades que o software proposto

deverá fornecer. Para a criação dos diagramas foi utilizada a ferramenta online de design app.diagram.net.

4.3.1 Diagrama De Caso De Uso

O diagrama de Caso de Uso descreve, principalmente, as interações que ocorrem entre os possíveis usuários do sistema e o próprio sistema. Logo, especifica um conjunto de funcionalidades relacionadas com os seus respectivos atores, ou seja, aqueles que utilizam essas funcionalidades (FOWLER, 2014). A Figura 1 apresenta o diagrama de casos de uso do sistema.

Figura 1 – Diagrama de Casos de Uso



Fonte: Elaborado pelo autor

Existem dois atores que se comunicam com os casos de uso. São eles:

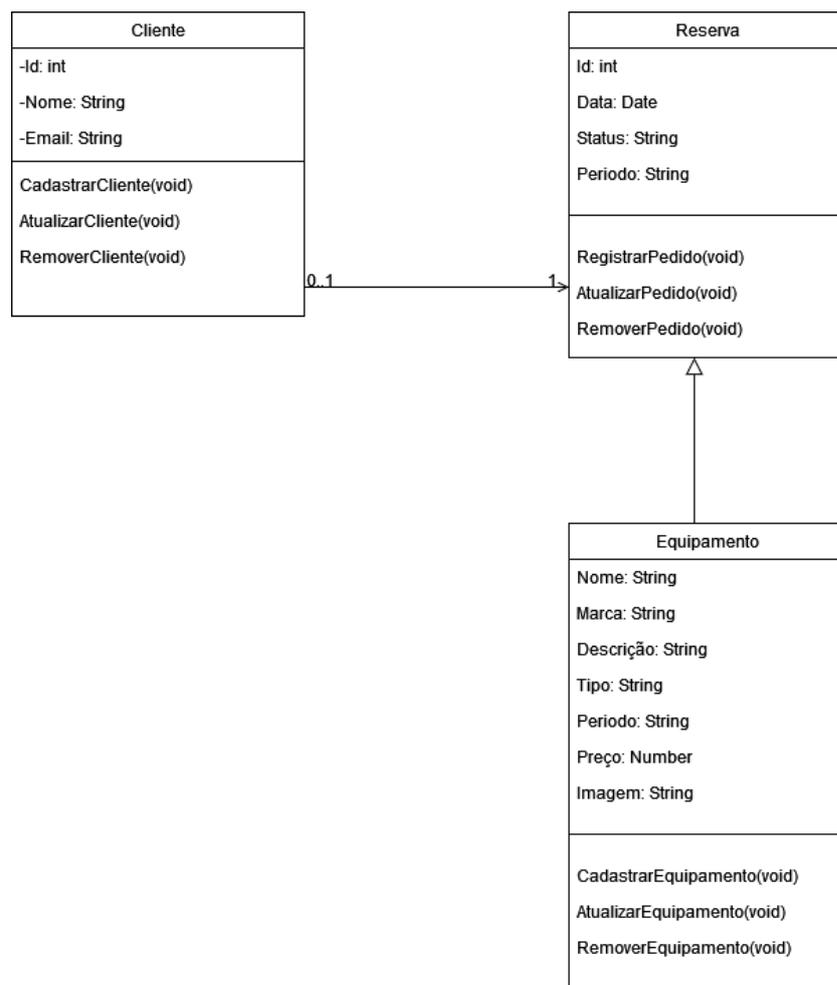
- **Administrador:** É o ator que se envolve com o sistema diretamente, sendo o principal responsável realizar tarefas como cadastrar equipamentos e operar a criação, consulta, atualização e destruição de dados de funcionarios, equipamentos e reservas.

- **Funcionário da Emergência:** É o ator que movimenta as ações do sistema. Compreende principalmente os colaboradores da área assistencial responsáveis pelos cuidados dos pacientes, que inclui principalmente enfermeiros e técnicos de enfermagem.

4.3.2 Diagrama De Classe

O diagrama de classe, possibilita visualizar as classes e atributos que irão compor o sistema, podemos considerar os atributos as informações que o objeto possui, já as operações podemos definir como aquelas em que o objeto pode realizar. entretanto, somente classes isoladas não fornecem informações de como tais objetos podem se comunicar, logo o diagrama de classe também prove a visualização dos relacionamentos entre classes, que são chamadas de associações e tem como objetivo descrever as informações que ocorrem entre classes. A [Figura 2](#) apresenta o diagrama de classe do sistema.

Figura 2 – *Diagrama de Classe*



Fonte: Elaborado pelo autor

4.4 Protótipos

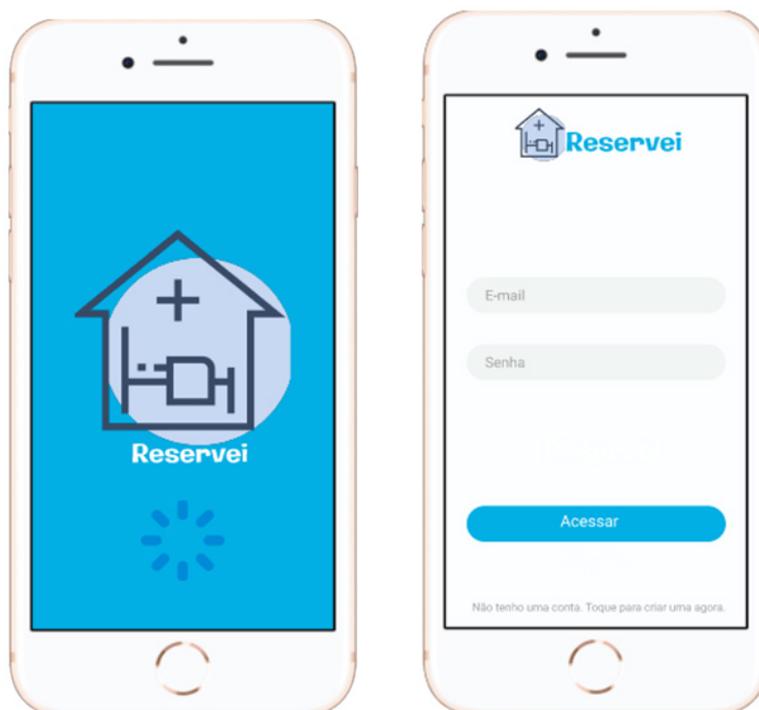
De acordo com [Pressman e Maxim \(2021\)](#) um protótipo é uma versão inicial de um sistema de software, usado para demonstrar conceitos, experimentar opções de projeto e descobrir mais sobre o problema e suas possíveis soluções. O desenvolvimento rápido e iterativo do protótipo é essencial para que os custos sejam controlados e os stakeholders do sistema possam experimentá-lo no início do processo de software.

Protótipos do sistema permitem aos usuários ver quão bem o sistema dá suporte a seu trabalho. Eles podem obter novas idéias para requisitos e encontrar pontos fortes e fracos do software; podem, então, propor novos requisitos do sistema. Além disso, o desenvolvimento do protótipo pode revelar erros e omissões nos requisitos propostos ([PRESSMAN, 2006](#)).

Para a criação dos protótipos foi utilizada a ferramenta FIGMA, que é uma gratuita e disponível online.

As [Figura 3](#), [Figura 4](#) e [Figura 5](#) mostra as telas de protótipos construídos durante a engenharia de requisitos.

Figura 3 – *Protótipos: Splash Screen e Login*

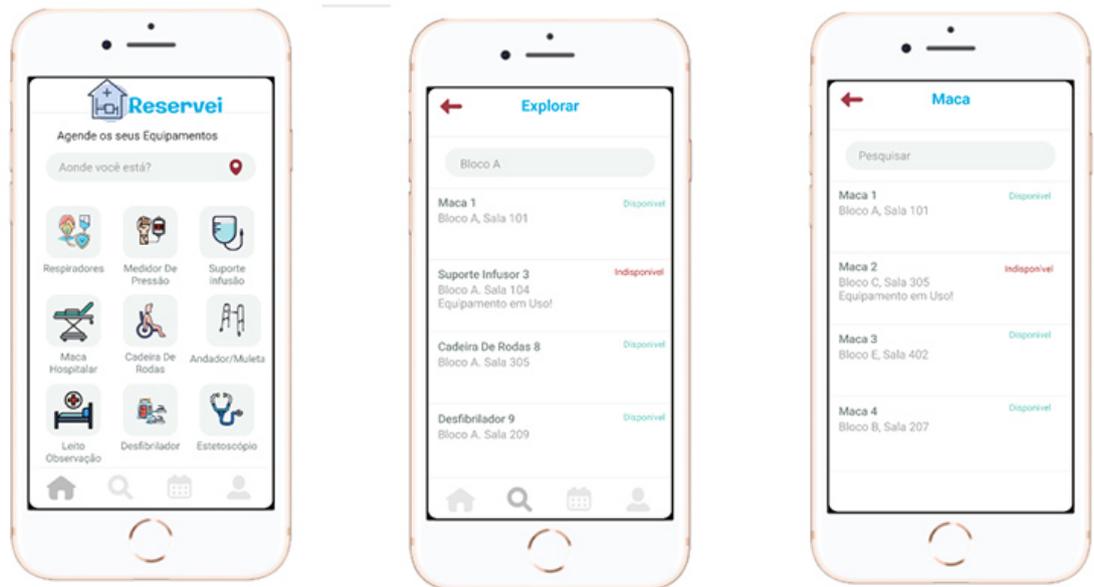


Fonte: Elaborado pelo autor

A tela splash tem como objetivo apresentar visualmente o produto enquanto o aplicativo carrega os serviços. A tela login tem a finalidade de autenticar os usuários para

acesso ao sistema.

Figura 4 – Protótipos: Home, Pesquisa e Categoria



Fonte: Elaborado pelo autor

A primeira experiência do usuário no aplicativo é importante, pois é neste momento que ele decide se o aplicativo será útil ou se irá desinstalá-lo. Portanto, o software deve ser intuitivo e de fácil entendimento para o usuário e deve apresentar as principais funções para auxiliar o usuário a decidir se este será útil ou não.

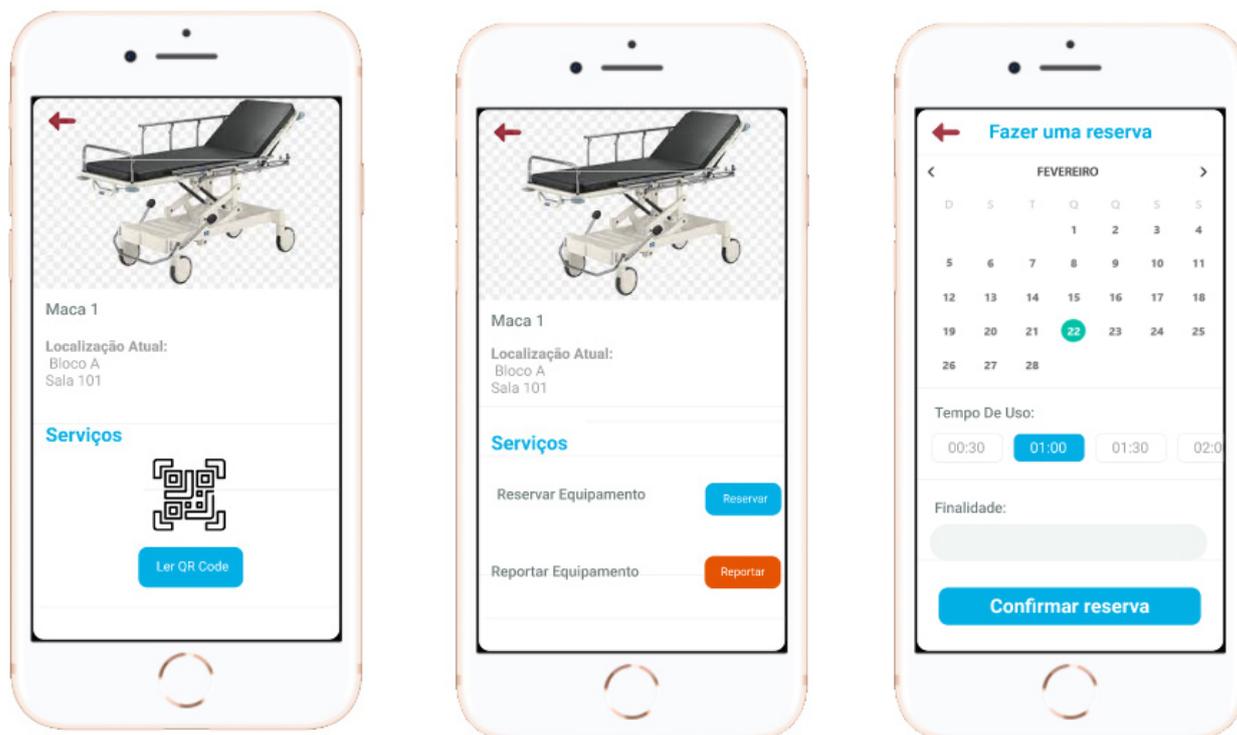
Portanto, a tela home contém os principais equipamentos utilizados nas emergências hospitalares para facilitar o acesso do usuário nas funções principais de pesquisa e localização e reserva.

Na tela Pesquisa, está disponível uma função que permite ao usuário buscar todos os equipamentos com base na sua localização. Essa tela irá retornar uma lista com todos os equipamentos que estão na localização informada.

A tela de Categoria apresenta uma lista completa de todos os equipamentos selecionados através de sua categoria na tela home, independente de sua localização. Nela são apresentados os equipamentos disponíveis para reserva e os indisponíveis que

eventualmente possam estar em uso no momento.

Figura 5 – *Protótipos: Reserva*

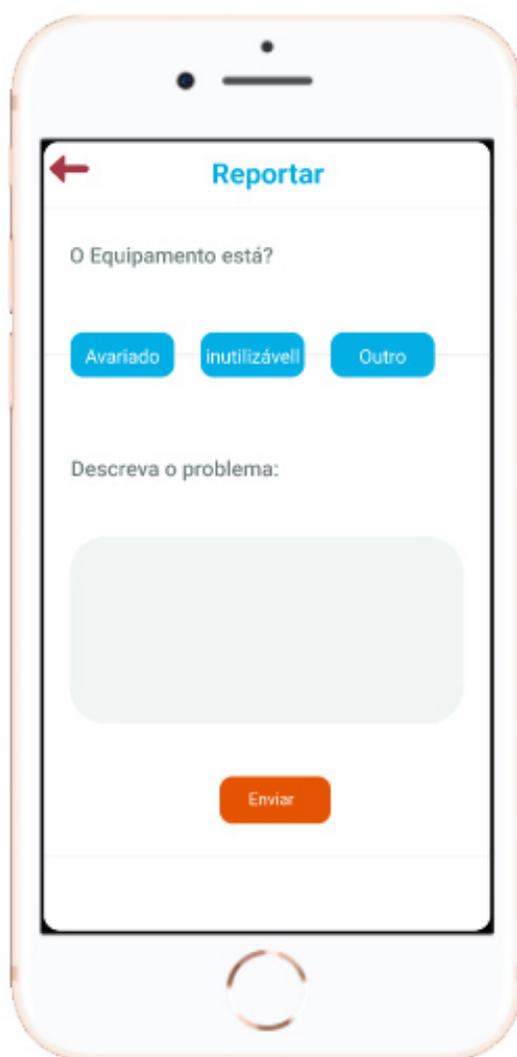


Fonte: Elaborado pelo autor

A primeira tela da [Figura 5](#) mostra informações do equipamento e também um botão para acessar os recursos do equipamento que só podem ser visualizados através da leitura do seu código QR Code. A segunda tela mostra os serviços disponíveis para aquele equipamento, a saber, efetuar uma reserva ou reportar algum problema encontrado. Por último a terceira tela efetiva a reserva, sendo possível selecionar o dia, tempo de permanência com o equipamento e para qual finalidade ele será reservado. A [Figura 6](#) é um dos serviços disponíveis após a leitura do código QR Code. Ela tem a finalidade de identificar avarias e demais problemas encontrados nos equipamentos, antes da reserva.

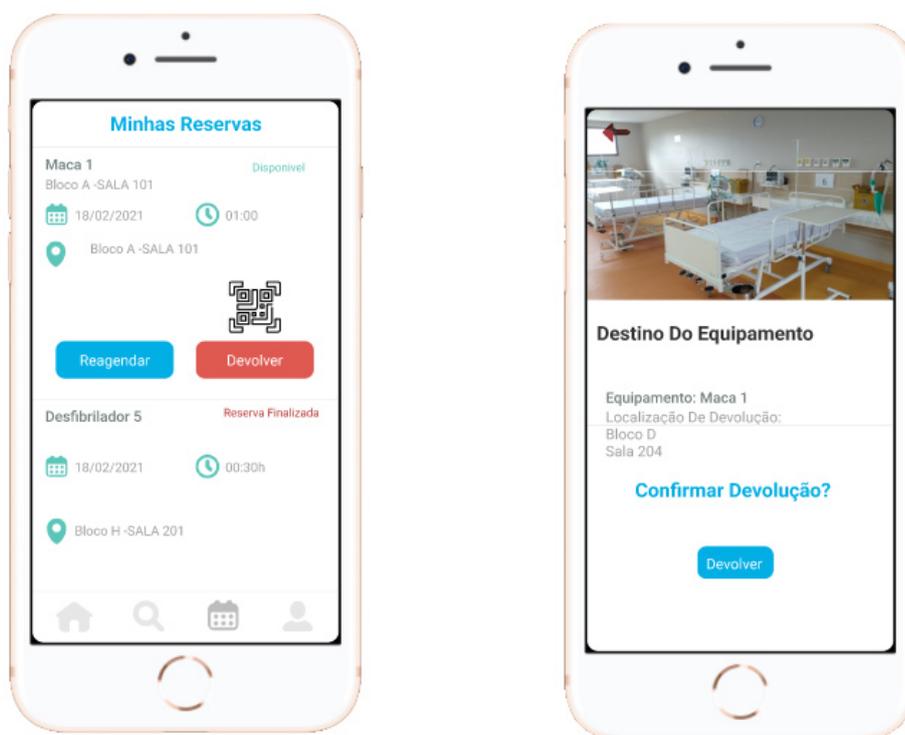
A [Figura 7](#) exibe todas as reservas já feitas pelo usuário como também a reserva em andamento, sendo possível ao usuário reagendar o equipamento e também devolvê-lo, mediante a leitura do código QR Code do lugar de devolução. A segunda tela da [Figura 6](#) mostra as informações do local de devolução e permite a confirmação deste equipamento, caso seja o local desejado.

Figura 6 – Protótipos: Report



Fonte: Elaborado pelo autor

Figura 7 – Protótipos: Reserva e Devolução



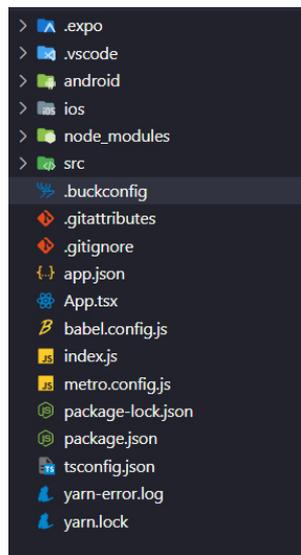
Fonte: Elaborado pelo autor

5 Implementação

5.1 Estrutura do Projeto

O projeto segue a seguinte hierarquia de pastas e arquivos (Figura 8):

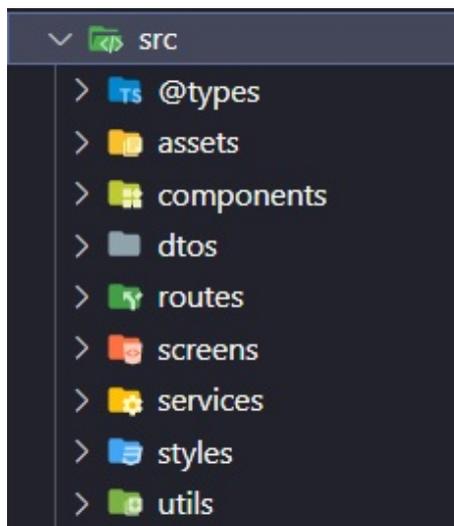
Figura 8 – Hierarquia de pastas e arquivos da aplicação



Fonte: Elaborado pelo autor

- **Expo:** A pasta se refere à configuração da ferramenta Expo para o desenvolvimento da aplicação.
- **Node Modules:** A pasta possui a coleção de módulos do Node JS utilizados na aplicação.
- **Source:** A pasta src possui todo o código fonte da aplicação, incluindo a configuração de conexão com o banco de dados e desenvolvimento completo do front-end.
- **App.js:** O arquivo é responsável por inicializar a aplicação.
- **Package.json:** Registra metadados importantes sobre o projeto que são necessários e também define atributos funcionais de um projeto que o Node Package Manager (NPM) usa para instalar dependências, executar scripts e identificar o ponto de entrada para o pacote.

Figura 9 – Hierarquia de pastas e arquivos da aplicação



Fonte: Elaborado pelo autor

Na pasta src está contida toda a aplicação desenvolvida, que será detalhada a seguir (Figura 9).

- **Assets:** A pasta possui a estilização de ícones e a imagem de carregamento da aplicação.
- **Components:** A pasta contém todos os componentes da aplicação e que geralmente são utilizados nas telas do aplicativo, incluindo botões, caixas de texto, entre outros.
- **Dtos:** A pasta apresenta uma interface responsável pelo tipo de objeto no banco de dados.
- **Routes:** A pasta possui os arquivos necessários para fazer a navegação entre telas da aplicação.
- **Screens:** A pasta contém todas as páginas/telas da aplicação.
- **Services:** A pasta engloba os arquivos de configuração de uma Application Programming Interface (API) com a aplicação.
- **Styles:** A pasta possui as configurações de estilo padronizadas que são usadas em todas as partes da aplicação.
- **Utils:** A pasta contém as configurações para utilizar dinamicamente os Scalable Vector Graphics (SVG), que são os ícones da aplicação.

5.2 API

API ou interface de programação de aplicativos basicamente interliga softwares, sistemas e aplicativos, possibilitando a comunicação e troca de informações, bem como cruzamento de dados entre os diferentes sistemas.

Com o objetivo de automatizar processos manuais, além da criação de novas funcionalidades, as APIs permitem que haja uma troca de informações e controles entre as aplicações, o que abre espaço para a criação de soluções feitas fora dos produtos. Existem APIs públicas, que são abertas para qualquer desenvolvedor ou empresa, e privadas, que são de uso exclusivo da organização. Para a implementação desse projeto foi utilizado a API `Json.server` que é utilizada para se fazer os testes e simular uma API de forma mais rápida.

JSON Server é um projeto simples que ajuda a configurar uma API REST com operações Create, Read, Update and Delete (**CRUD**), muito rapidamente (<https://github.com/typicode/json-server>).

A [Figura 10](#) ilustra um arquivo que contém os dados que devem ser exibidos pela API REST. Para objetos contidos na estrutura JSON, os endpoints CRUD são criados automaticamente.

Figura 10 – Arquivo JSON

```
equipos: [
  {
    "id": "5508460-7231-48ac-8b02-7d8be7c8d0d7",
    "brand": "Marca Hospitalar",
    "name": "Marca",
    "about": "Marca é um tipo de caso, assentada numa armação metálica e articulada, com",
    "period": "A hora",
    "price": "Indisponível"
  },
  {
    "type": "Hybrid_wator",
    "thumbnail": "https://uf.pepinq.com/peg/682/31/peg-transparent-hospital-bed-c",
    "accessories": [
    ],
    "photos": [
      "https://uf.pepinq.com/peg/682/31/peg-transparent-hospital-bed-cama-electri",
      "https://uf.pepinq.com/peg/682/31/peg-transparent-hospital-bed-cama-electri",
      "https://uf.pepinq.com/peg/682/31/peg-transparent-hospital-bed-cama-electri"
    ]
  }
]
```

Fonte: Elaborado pelo autor

A [Figura 9](#) mostra um objeto do tipo equipamento e seus respectivos atributos, sendo eles:

- **id: Tipo - string:** É uma chave gerada aleatoriamente.
- **brand: string;** Representa a marca do equipamento.
- **name: string;** Representa o nome do equipamento.
- **about: string;** Apresenta uma breve descrição do equipamento.
- **period: string;** Apresenta um período de reserva do equipamento
- **price: number;** Representa o preço do equipamento.

- **type: string;** Define o tipo do equipamento.
- **thumbnail: string;** Apresenta uma imagem ilustrativa do equipamento.
- **accessories:** Vetor de string, contém uma lista de acessórios que acompanham o equipamento.

5.2.1 Conexão com o API

A conexão com o banco de dados utiliza uma série de parâmetros fornecidos pelo Json server. Para que essa conexão seja estabelecida com o banco da dados, foi necessário a instalação de pacotes no Node JS, incluindo o próprio json server (Figura 11).

Figura 11 – *Conexão com API*

```
"scripts": {  
  "start": "expo start --dev-client",  
  "android": "expo run:android",  
  "ios": "expo run:ios",  
  "web": "expo start --web",  
  "api": "json-server ./src/services/server.json --host 192.168.2.16 --port 3333 - delay"  
},
```

Fonte: Elaborado pelo autor

Por padrão, o Json server retorna o localhost, o que se torna um problema quando se utiliza o Expo para o desenvolvimento nativo. A fim de facilitar essa conexão, foi criado um script no package.json para prover a conexão com a API que basicamente executa o json server, passa o endereço do arquivo e o endereço Internet Protocol (IP) da máquina e também uma porta para evitar conflitos além de um tempo de espera (delay) para simular o carregamento de uma Application Programming Interface (API).

5.2.2 Entrada de Dados

Para obter os dados da API e consumi-los, foi necessária a instalação do axios, que é uma biblioteca para tratamento de requisições http. Foi criado um arquivo API em services para centralizar as requisições da aplicação (Figura 12).

O baseUrl é a informação da API que é fixa, no caso o endereço da API já definido anteriormente, por fim exporta-se a API para importar na aplicação.

5.2.3 Saída de Dados

Para retornar as informações de uma API é preciso considerar que pode levar algum tempo, pois existe fatores como taxa de transmissão da internet e a conexão. Nesse contexto foi necessário criar uma função assíncrona. Para isso, foi utilizado o async caso

Figura 12 – Conexão com API: Axios

```
src > services >  api.ts > ...  
1   import axios from 'axios';  
2  
3   const api = axios.create({  
4     |   baseURL: 'http://192.168.2.16:3333',  
5   | });  
6  
7   export {api};
```

Fonte: Elaborado pelo autor

a aplicação não tenha tido tempo para obter a resposta da API. Com isso, foi possível utilizar uma função chamada "await" para garantir que a aplicação aguarde uma resposta da API (Figura 13).

Figura 13 – Conexão com API: Assíncrona

```
useEffect(() => {  
  async function fetchEquip() {  
    try{  
      const response = await api.get('/equips');  
      setEquip(response.data);  
    }catch(error){  
      console.log(error);  
    }finally{  
      setLoading(false);  
    }  
  }  
  fetchEquip();  
}, []);
```

Fonte: Elaborado pelo autor

5.3 Fluxo do aplicativo

A Figura 14 mostra a tela inicial do aplicativo.

Figura 14 – Tela Inicial



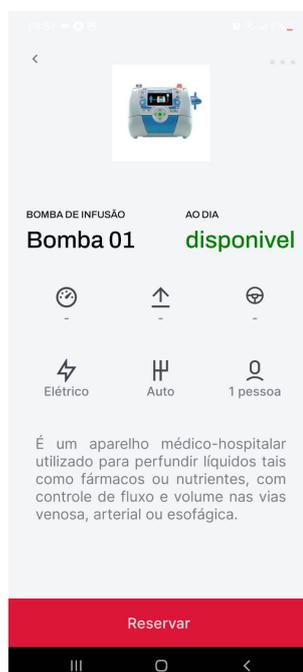
Fonte: Elaborado pelo autor

A tela inicial tem o intuito de apresentar os equipamentos médicos hospitalares móveis mais comuns, sejam eles disponíveis ou indisponíveis. O usuário tem acesso a alguns detalhes do equipamento e pode realizar a reserva do equipamento após clicar sobre o ícone do equipamento. Para melhorar a usabilidade e eficiência do aplicativo, foi utilizado um componente do React Native chamado ScrollView, que é um contêiner de rolagem genérico para permitir a visualização de vários componentes. Com isso, mantém-se na tela um número razoável de equipamentos por vez, para não gerar uma sobrecarga cognitiva no usuário. Os itens roláveis podem ser heterogêneos e é possível rolar vertical e horizontalmente. Essa abordagem é recomendada quando tem-se uma lista muito grande de itens, pois proporciona a rolagem de tela e também auxilia no carregamento de dados da aplicação.

A [Figura 15](#) mostra a tela de detalhes de um equipamento, a qual retorna uma breve descrição sobre ele e alguns dos principais atributos, Nela também há alguns componentes como o botão que volta para a tela anterior, e um componente que faz a rota para a tela de agendamentos que é apresentada a seguir ([Figura 16](#)).

A tela de reservas ([Figura 16](#)) tem o objetivo de definir um período de reserva de um equipamento. Ela também trata reservas inválidas, caso o usuário não insira nenhum dia ou escolha dias anteriores ao seu dia atual. Dada a dinâmica do atendimento nas emergências, espera-se que a maioria das reservas sejam imediatas, ou seja, na data e

Figura 15 – Tela De detalhes de Um Equipamento



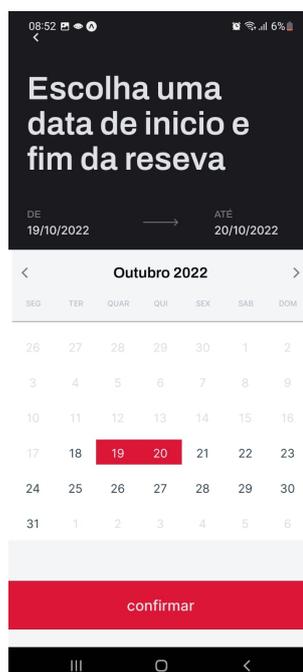
Fonte: Elaborado pelo autor

horário correntes.

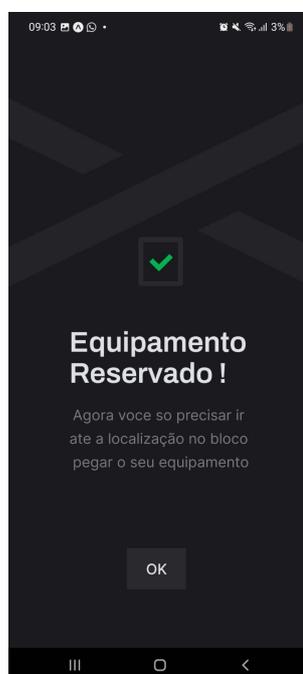
A tela de detalhes de uma reserva (Figura 17), fornece todos os detalhes de um objeto, como também os detalhes do processo de reserva daquele objeto, a fim de permitir que o usuário possa verificar novamente se é realmente aquele objeto que deseja reservar e se o período da reserva está correto antes de finalmente efetuar a reserva. Essa tela também possui três componentes: Botão que permite ao usuário voltar para a tela anterior, um componente slider, que permite ao usuário deslizar e visualizar mais imagens daquele equipamento e por fim um botão que efetiva a reserva do equipamento.

Figura: Tela de Reserva Finalizada

Figura 16 – Tela de Reserva



Fonte: Elaborado pelo autor



Fonte: Elaborado pelo autor (2022)

Por ultimo , a tela de reserva finalizada, tem o objetivo de demonstrar ao usuario que a reserva foi efetuada com sucesso, nela temos um component bottom que permite ao usuario fechar a mensagem e voltar a tela inicial da aplicação

Figura 17 – Tela de detalhes da reserva



Fonte: Elaborado pelo autor

6 Conclusão

Este trabalho apresentou todo o processo de desenvolvimento de um aplicativo para gerenciamento do fluxo de equipamentos médicos móveis nas emergências hospitalares, onde foi possível acompanhar desde o processo de concepção de negócio até a implementação do aplicativo.

Os equipamentos médicos são componentes fundamentais dos serviços de saúde, em especial nas emergências. A indisponibilidade dos equipamentos quando estes forem necessários em um atendimento pode ter consequências graves para a saúde do paciente. Nesse contexto, foi proposto um aplicativo de gerenciamento do fluxo de equipamentos móveis para auxiliar na redução de desperdícios relacionados ao seu uso nas emergências. Uma gestão eficiente e ágil dos equipamentos móveis nas emergências hospitalares pode melhorar a assistência aos pacientes, além de evitar conflitos e desgaste nos profissionais de saúde que atuam por longos períodos nas unidades.

6.1 Trabalhos Futuros

Com o intuito de que os administradores tenham a possibilidade de cadastrar, equipamentos e usuários, está sendo desenvolvida uma interface World Wide Web ([WEB](#)) a fim também de tornar mais simples a geração de códigos Quick Response Code ([QR Code](#)) e relatórios. Esta proposta também consiste na criação de um banco de dados centralizado em nuvem para centralizar as transações que ocorrem nos aplicativos dos usuários. Assim, usuários que optassem por utilizar contas, teriam backup de seus dados em nuvem, não precisando se preocupar com desinstalações do aplicativo, furtos, danificações do aparelho e entre outros.

Referências

- BULL, F. C. et al. World health organization 2020 guidelines on physical activity and sedentary behaviour. *British Journal of Sports Medicine*, BMJ, v. 54, n. 24, p. 1451–1462, nov. 2020. Disponível em: <<https://doi.org/10.1136/bjsports-2020-102955>>. Citado na página 16.
- CHAROENSIRIWATH, C. et al. Applying qr code and mobile application to improve service process in thai hospital. In: IEEE. *2015 12th International Joint Conference on Computer Science and Software Engineering (JCSSE)*. [S.l.], 2015. p. 114–119. Citado na página 23.
- CNN, B. *Brasil tem mais smartphones que habitantes, aponta FGV*. 2022. Url<https://www.cnnbrasil.com.br/business/brasil-tem-mais-smartphones-que-habitantes-aponta-fgv/>. Citado na página 15.
- DEVMEDIA. *Introdução ao Visual Studio Code*. 2016. Url<https://www.devmedia.com.br/introducao-ao-visual-studio-code/34418>. Citado na página 28.
- FOWLER, M. *UML Essencial: um breve guia para linguagem padrão*. [S.l.]: Bookman editora, 2014. Citado na página 33.
- GARTNER, A. *Gartner Says Worldwide Sales of Smartphones Grew 7 Percent in the Fourth Quarter of 2016*. 2017. Url<https://www.gartner.com/en/newsroom/press-releases/2017-02-15-gartner-says-worldwide-sales-of-smartphones-grew-7-percent-in-the-fourth-quarter-of-2016>. Citado na página 26.
- GERHARDT, S. *Métodos de pesquisa*. [S. l, 2009. Citado na página 25.
- GUEDES, G. T. *Uml 2. Uma Abordagem Prática”, São Paulo, Novatec, 2009*. Citado na página 32.
- KHAWAS, C.; SHAH, P. Application of firebase in android app development-a study. *International Journal of Computer Applications*, v. 179, n. 46, p. 49–53, 2018. Citado na página 28.
- LANCMAN, S. et al. Os trabalhadores do contexto hospitalar em tempos de pandemia: singularidades, travessias e potencialidades. *Interface - Comunicação, Saúde, Educação*, FapUNIFESP (SciELO), v. 25, n. suppl 1, 2021. Disponível em: <<https://doi.org/10.1590/interface.210119>>. Citado na página 14.
- MATOS, B. R. D.; BRITTO, J. G. de. Estudo comparativo entre o desenvolvimento de aplicativos móveis utilizando plataformas nativas e multiplataforma. 2016. Citado 2 vezes nas páginas 26 e 27.
- MORONEY, L. The firebase realtime database. In: *The Definitive Guide to Firebase*. [S.l.]: Springer, 2017. p. 51–71. Citado na página 28.

MORSCH, D. J. A. *COMO ADQUIRIR EQUIPAMENTOS MÉDICOS HOSPITALARES E SUA MANUTENÇÃO?* 2018.

Url<https://telemedicinamorsch.com.br/blog/equipamentos-medicos-hospitalares>.

Citado 2 vezes nas páginas 19 e 20.

MOTTA, M. V. da; MENA, H.; PIACSEK, G. Urgência e emergência. os conceitos frente às normas administrativas e legais e suas implicações na clínica médica.

Saúde Ética & Justiça, Universidade de Sao Paulo, Agencia USP de Gestao da Informacao Academica (AGUIA), v. 22, n. 2, p. 81–94, dez. 2017. Disponível em:

<<https://doi.org/10.11606/issn.2317-2770.v22i2p81-94>>. Citado na página 18.

MOZILLA, F. *Foundation, Mozilla*. 2022. Url<https://foundation.mozilla.org/pt-BR/>.

Citado na página 27.

NATIVE, R. React native. *línea*]. Disponível em: <https://reactnative.dev/>. [Último acesso: 2 de novembro 2019], 2020. Citado na página 27.

NÚÑEZ, C.; CASTRO, D. Management information system of medical equipment using mobile devices. In: IOP PUBLISHING. *Journal of Physics: Conference Series*. [S.l.], 2011. v. 313, n. 1, p. 012003. Citado na página 24.

OVERFLOW, S. *Developer Survey*. 2021.

Url<https://insights.stackoverflow.com/survey/2021most-popular-technologies>.

Citado na página 28.

PEDROZO, O. *QUAL A IMPORTÂNCIA DA CONCEPÇÃO NO DESENVOLVIMENTO DE SOFTWARE?* 2018. Url<https://softdesign.com.br/blog/qual-importancia-da-concepcao-e-anteprojeto-no-desenvolvimento-de-software>. Citado na página 30.

PETE, H. Why did we build react. *línea*]. Disponível em: <https://reactjs.org/blog/2013/06/05/why-react.html>. [Último acesso: 2 de junho 2022], 2017. Citado na página 27.

PIRES, D. Reestruturação produtiva e trabalho em saúde no brasil. In: *Reestruturação produtiva e trabalho em saúde no Brasil*. [S.l.: s.n.], 2008. p. 253–253. Citado na página 14.

PITTA, A. Hospital: dor e morte como ofício. In: *Hospital: dor e morte como ofício*. [S.l.: s.n.], 1990. p. 198–198. Citado na página 14.

PRESSMAN, R. *Engenharia de Software, McGrawHill, 6a*. [S.l.]: Edição, 2006. Citado 2 vezes nas páginas 26 e 35.

PRESSMAN, R. S.; MAXIM, B. R. *Engenharia de software-9*. [S.l.]: McGraw Hill Brasil, 2021. Citado 4 vezes nas páginas 30, 31, 32 e 35.

SILVA, A. L. A. et al. Produção de subjetividade e gestão em saúde: cartografias da gerência. 2004. Citado na página 14.

SOMMERVILLE, I. Software engineering 9th edition. *ISBN-10*, v. 137035152, p. 18, 2011. Citado na página 25.

TENHUNEN, S. Expo react native-sovellusten kehitysalustana. 2021. Citado na página 29.

WEBER, L.; GRISCI, C. L. I. Trabalho, gestão e subjetividade: dilemas de chefias intermediárias em contexto hospitalar. *Cadernos EBAPÉ.BR*, FapUNIFESP (SciELO), v. 8, n. 1, p. 53–70, mar. 2010. Disponível em: <<https://doi.org/10.1590/s1679-39512010000100005>>. Citado na página 15.

WU, W. React native vs flutter, cross-platforms mobile application frameworks. Metropolia Ammattikorkeakoulu, 2018. Citado 2 vezes nas páginas 26 e 27.

YU, R.; DUAN, X.; JIAO, B. Design and implement of mobile equipment management system based on qrcode. In: IOP PUBLISHING. *Journal of Physics: Conference Series*. [S.l.], 2017. v. 887, n. 1, p. 012036. Citado na página 23.