INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO GRANDE DO NORTE

FERNANDA DA SILVA CALDAS

ANÁLISE DOS DESAFIOS DA ELICITAÇÃO DE REQUISITOS EM UM HACKATHON: UM ESTUDO DE CASO COM MARCASUS

NATAL

2025

FERNANDA DA SILVA CALDAS

ANÁLISE DOS DESAFIOS DA ELICITAÇÃO DE REQUISITOS EM UM HACKATHON: UM ESTUDO DE CASO COM MARCASUS

Trabalho de Conclusão apresentado ao Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, em cumprimento às exigências legais como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof.º Me. Fabiano Papaiz

NATAL

2025

Caldas, Fernanda da Silva.
C145a Análise dos desafios da elicitação de requisitos em um hackathon : um estudo de caso com marcasus / Fernanda da Silva Caldas. – 2025.
40 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, Natal, 2025.
Orientador(a): Me. Fabiano Papaiz.

1. Engenharia de requisitos. 2. Elicitação de requisitos. 3. Sistema Único de Saúde (Brasil). 4. Hackathon. I. Título.

SIBi/IFRN CDU: 62

Sistema Integrado de Bibliotecas
Elaborada pela Bibliotecária
Leticia P. A. de Sousa – CRB-15/1018

FERNANDA DA SILVA CALDAS

ANÁLISE DOS DESAFIOS DA ELICITAÇÃO DE REQUISITOS EM UM HACKATHON: UM ESTUDO DE CASO COM MARCASUS

Trabalho de Conclusão apresentado ao Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, em cumprimento às exigências legais como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Aprovado em 29 de julho de 2025.

BANCA EXAMINADORA

Fabiano Papaiz, Prof. Me. – Orientador

Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte

Marília Aranha Freire, Prof.ª Dr.ª

Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte

Demóstenes Santos de Sena, Prof. Dr.

Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte

AGRADECIMENTOS

Há jornadas que não se medem em páginas e não existem palavras para expressar minha gratidão ao grupo de pessoas que me incentivou até aqui, mas tentarei.

Aos meus amigos, otimistas, serei sempre grata por vocês terem acreditado que eu chegaria até aqui, especialmente quando todas as circunstâncias pareciam dizer que não.

À turma que me acolheu no retorno à graduação, pelas brincadeiras em sala, pelos grupos de estudo, vocês não têm ideia de como tornaram os últimos e quase intermináveis semestres mais leves, por muitas vezes me motivaram sem saber.

A Lucas Nithael, o verdadeiro MVP, que me salvou em incontáveis entregas de PDS e estava sempre por perto com uma palavra de incentivo, muito obrigada, meu amigo! E aos meninos do SUGEV, que foram meu suporte técnico, equipe de logística e serviço de carona particular. Vocês foram essenciais.

À Professora Marília Aranha Freire, por ter praticado a arte da insistência nobre, minha eterna gratidão. Depois de quase três anos, foi seu apoio que me deu a segurança para retomar, eu já tinha me conformado com meu destino de "quase analista de sistemas", mas você não.

À minha mãe e à minha família, por respeitarem meu tempo e minhas decisões, inclusive quando decidi pausar a graduação.

E, de forma muito especial, ao meu pai, Dagoberto Sthenio (*in memoriam*). Ele sempre me ensinou o valor da educação e o quanto os estudos eram importantes. Eu realmente gostaria que ele estivesse aqui para compartilhar este momento comigo.

Por fim, preciso agradecer à pessoa mais difícil de convencer: eu mesma. Por finalmente ter cedido à pressão... de mim mesma. Foi um passeio de muitas brigas internas, mas, no fim, fizemos as pazes. E vencemos.

RESUMO

Este trabalho analisa as adaptações da elicitação de requisitos no desenvolvimento de um protótipo funcional de um sistema idealizado em um *hackathon*, por meio de um estudo de caso sobre o MarcaSUS, uma solução projetada para otimizar o agendamento de consultas no Sistema Único de Saúde (SUS) do Rio Grande do Norte. O objetivo geral é analisar o processo de elicitação em um cenário com restrições de tempo, buscando entender as estratégias da equipe diante da impossibilidade de aplicar um ciclo completo de levantamento de requisitos, revelando que a autora recorreu a uma abordagem pragmática, combinando entrevistas informais e, principalmente, *brainstorming*. Conclui-se que, mesmo em um ambiente não ideal, é possível aplicar os fundamentos da Engenharia de Requisitos de forma adaptada, e que a flexibilidade metodológica é importante para o sucesso de projetos com limitações rigorosas. A experiência documentada contribui para a compreensão da elicitação de requisitos em contextos ágeis e de prototipagem rápida.

Palavras-chave: Engenharia de Requisitos, Elicitação de Requisitos, Hackathon, SUS.

ABSTRACT

This paper analyzes the adaptations of requirements elicitation in the development of a functional prototype of a system conceived during a hackathon, through a case study on MarcaSUS, a solution designed to optimize appointment scheduling in the Rio Grande do Norte's Health System (SUS). The main objective is to analyze the elicitation process in a time-constrained scenario, seeking to understand the team's strategies when faced with the impossibility of applying a complete requirements gathering cycle, revealing that the author resorted to a pragmatic approach, combining informal interviews and, predominantly, brainstorming. It is concluded that, even in a non-ideal environment, it is possible to apply the fundamentals of Requirements Engineering in an adapted manner, and that methodological flexibility is essential for the success of projects with strict limitations. The documented experience contributes to the understanding of requirements elicitation in agile and rapid prototyping contexts.

Keywords: Requirements Engineering, Requirements Elicitation, Hackathon, SUS.

LISTA DE FIGURAS

Figura 1 – Engenharia de Requisitos não é linear 16
Figura 2 – Etapas típicas do processo de desenvolvimento de um software 18
Figura 3 – Fluxo da jornada do paciente no agendamento de consultas 23
Figura 4 – Processo de software adaptado pela equipe 24
Figura 5 – Diagrama de caso de uso do paciente 25
Figura 6 – Diagrama de caso de uso do funcionário 26
Figura 7 – Tela de login da plataforma web com validação de CPF 26
Figura 8 – Atualizar Agenda - Busca por médico via nome e CRM 27
Figura 9 – Exibição e atualização da agenda médica 27
Figura 10 – Seleção de paciente para marcação de consulta (“Cadastrar Consulta”) 28
Figura 11 – Seleção do médico para conclusão da marcação da consulta 28
Figura 12 – Elicitação no cotidiano *versus* elicitação no *hackathon* 32

LISTA DE TABELAS

Tabela 1 – Requisitos funcionais do sistema 22
Tabela 2 – Comparativo entre o Processo de Elicitação Ideal e o Praticado no *Hackathon*..... 33
Tabela 3 – Análise Comparativa dos Princípios da ER no Contexto do Hackathon..... 35

LISTA DE SIGLAS

API	Interface de Programação de Aplicações
ER	Engenharia de Requisitos
IFRN	Instituto Federal do Rio Grande do Norte
IOS	Sistema Operacional do iPhone
MVP	Produto Mínimo Viável
PNS	Pesquisa Nacional de Saúde
RF	Requisito Funcional
RNF	Requisito Não Funcional
SUS	Sistema Único de Saúde
TADS	Tecnologia em Análise e Desenvolvimento de Sistemas
UBS	Unidade Básica de Saúde
UML	Linguagem de Modelagem Unificada

SUMÁRIO

1 INTRODUÇÃO 12
1.1 OBJETIVOS 13
1.1.1 Objetivo geral 13
1.1.2 Objetivos específicos 13
2 FUNDAMENTAÇÃO TEÓRICA 14
2.1 ENGENHARIA DE REQUISITOS 14
2.1.1 Elicitação de requisitos 15
2.1.2 Técnicas de elicitação 17
2.2 LINGUAGEM DE MODELAGEM UNIFICADA 18
2.3 PROCESSO DE SOFTWARE 18
2.4 HACKATHONS 19
2.5 TECNOLOGIAS UTILIZADAS 19
2.5.1 Tecnologias de suporte 19
2.5.2 Tecnologias de desenvolvimento 19
3 METODOLOGIA 20
3.1 O HACKATHON DA LOGAP 20
3.2 COLETA DE DADOS 20
3.2.1 Entendimento do processo de marcação de consultas 20
3.2.2 Resultado das entrevistas com os sujeitos 22
3.2.3 Resultado da entrevista com médica de uma UBS 23
3.3 PROCESSO DE DESENVOLVIMENTO ADOTADO 24
4 RESULTADOS 25
4.1 MARCASUS 25
4.1.1 Casos de Uso 25
4.1.2 Plataforma *Web* 26
4.2 ESTUDO DE CASO - DESAFIOS DA ELICITAÇÃO NO *HACKATHON* 28
4.2.1 Análise da elicitação de requisitos no cotidiano da autora 29
4.2.2 Análise dos principais desafios da elicitação no *hackathon* 30
4.2.3 Análise das técnicas adotadas 31
4.2.4 Análise da comparação do Processo Ideal vs. Processo Adaptado 32
4.2.5 Benefícios da Engenharia de Requisitos (ER) 33
4.2.6 Análise da aplicação dos princípios da ER no cenário do *hackathon* 34

SUMÁRIO

REFERÊNCIAS 39

1 INTRODUÇÃO

Dentro do ciclo de vida da Engenharia de Requisitos (ER), a elicitação[1] destaca-se como a fase inicial e mais crítica (Belgamo; Martins, 2000). Ela envolve o processo de transformar os desejos implícitos do cliente em requisitos explícitos, compreensíveis e verificáveis para o sistema a ser desenvolvido (Glinz et al., 2010). É um processo iterativo e colaborativo, que exige a aplicação de técnicas específicas e uma intensa interação entre analistas e usuários (Barbosa et al., 2009). A elicitação não apenas dá início ao desenvolvimento do *software*, mas também estabelece a base para as atividades de análise, especificação, verificação e gerenciamento dos requisitos. Esse processo é responsável por descobrir, analisar e validar as necessidades dos *stakeholders*, tratando-se de uma tarefa complexa, que demanda tempo e objetividade.

Desta feita, o cenário ideal para a elicitação envolve tempo, colaboração e uma validação contínua, elementos que asseguram a qualidade da especificação. Entretanto, o diferencial deste projeto está em seu contexto de origem: o MarcaSUS foi idealizado durante um *hackathon* realizado em aproximadamente dois dias, um ambiente marcado por prazos curtos, decisões rápidas e escassez de cerimônias formais.

Diante do exposto, este trabalho é norteado pelo desafio no processo de elicitação de requisitos quando realizado no ambiente restritivo de um *hackathon*, apresentando os conceitos fundamentais da engenharia de requisitos e da elicitação, descrevendo a experiência da autora no desenvolvimento do sistema no contexto do evento. Tratando-se de uma pesquisa de natureza qualitativa e exploratória.

Desse modo, além deste capítulo introdutório, este trabalho é composto por mais quatro capítulos, que organizam o conteúdo. Iniciamos com a introdução, com a contextualização do tema, os objetivos do estudo e a estrutura do trabalho. Em seguida, é abordada a fundamentação teórica, que fornece o embasamento necessário para a execução deste trabalho. No capítulo 3 é apresentado a metodologia, com a descrição das fases da coleta de dados da pesquisa. Por fim, os capítulos finais reúnem os resultados obtidos através do estudo de caso, as considerações finais e as referências utilizadas, ressaltando as contribuições do trabalho e os benefícios adquiridos do seu processo.

[1] A palavra vem do latim *elicitare*, que significa "fazer sair", "provocar" ou "trazer à tona".

1.1 OBJETIVOS

1.1.1 Objetivo geral

Analisar o processo de elicitação de requisitos no desenvolvimento de sistemas em um ambiente caracterizado pela restrição de tempo, por meio de um estudo de caso realizado no MarcaSUS, para identificar os principais desafios e a técnica mais adequada para o grupo, neste contexto.

1.1.2 Objetivos específicos

a. Analisar a elicitação de requisitos realizada sob as restrições do evento, detalhando a metodologia utilizada para o levantamento desses requisitos necessários;
b. Realizar uma comparação do Processo Ideal *versus* o Processo Realizado;
c. Analisar a técnica mais adequada de elicitação de requisitos em contextos como *hackathons* baseado na experiência.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta uma pesquisa bibliográfica realizada para embasar e contextualizar o presente estudo, destacando conceitos e abordagens para seu desenvolvimento.

2.1 ENGENHARIA DE REQUISITOS

A ER é uma área dentro da engenharia de *software* fundamental no desenvolvimento de sistemas, regida por um conjunto de princípios, ela contempla atividades relacionadas à definição e gerência dos requisitos, onde dentro dessas atividades podemos destacar: elicitar, analisar, documentar, validar (os requisitos) e, durante todo o desenvolvimento de *software*, gerenciá-los (Pfleeger, 2004).

O conjunto de princípios que regem a ER aplica-se a todas as suas tarefas, atividades e práticas, conforme Glinz et al. (2010), esses termos se distinguem da seguinte forma: uma **tarefa** é um pedaço coerente de trabalho a ser feito (ex: elicitar), uma **atividade** é uma ação ou conjunto de ações que uma pessoa ou grupo executa para realizar a tarefa (ex: identificar *stakeholders*) e **prática** se caracteriza como uma forma comprovada de como realizar certas atividades ou tarefas (ex: escolha da técnica de elicitação). Dados esses conceitos, os princípios da ER são:

1. **Orientação para o valor:** os requisitos são um meio para atingir um fim, e não o fim em si. O foco deve ser sempre no valor que o sistema trará para o negócio e para os usuários;
2. ***Stakeholders***: a ER é sobre satisfazer os desejos e necessidades dos *stakeholders*, eles são a fonte primária e o destino de todo o trabalho;
3. **Entendimento comum**: o desenvolvimento bem-sucedido de sistemas é impossível sem uma base de compreensão comum compartilhada entre todos os envolvidos;
4. **Contexto**: sistemas não podem ser compreendidos isoladamente. É crucial entender o contexto em que o sistema irá funcionar;
5. **Problema - Requisito - Solução**: uma trinca inevitavelmente entrelaçada entre o problema a ser resolvido, os requisitos que descrevem a solução e a solução em si;
6. **Validação**: requisitos não validados são inúteis, é essencial verificar se a especificação segue o que o cliente realmente precisa;
7. **Evolução**: a mudança de requisitos não é um acidente, mas sim uma característica inerente ao desenvolvimento de sistemas;

8. **Inovação**: em um mercado competitivo, mais do mesmo não é mais o suficiente, a ER deve ser uma porta de entrada para a inovação;
9. **Trabalho sistemático e disciplinado**: não podemos prescindir deles na ER para garantir qualidade e consistência.

É importante destacar que, no âmbito da engenharia de requisitos, existem alguns conceitos fundamentais que podem gerar confusão entre analistas em formação, especialmente no que diz respeito à distinção entre Requisitos Funcionais e Não Funcionais.

Desde o início do curso de Tecnologia em Análise e Desenvolvimento de Sistemas (TADS) do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte (IFRN), aprendemos que os Requisitos Funcionais (RF) descrevem o comportamento do sistema e o que ele deve fazer. Eles especificam como o sistema deve reagir a determinadas entradas e como deve se comportar em cenários específicos. Sommerville (2011) afirma que a ausência de um requisito funcional implica que o sistema não precisa executar aquela função. Por outro lado, os Requisitos Não Funcionais (RNF) referem-se às restrições e qualidades que o sistema deve atender durante sua operação. Conforme destaca Fagundes (2011, p. 9), "Requisitos não funcionais não alteram a definição do comportamento da solução, mas definem outros aspectos relevantes para a aplicação como meios de exibição, performance, acessibilidade etc."

Esses requisitos não especificam diretamente *o que* o sistema deve fazer, mas como ele deve se comportar ou sob quais condições deve funcionar. Analisando conceitualmente e conforme aponta a literatura, essa divisão é essencial, pois ambos os tipos de requisitos têm propósitos distintos e são validados de maneiras diferentes. Por conseguinte, as definições de requisitos desempenham um papel fundamental na engenharia de requisitos, oferecendo um conjunto de boas práticas para auxiliar na definição eficiente de sistemas e suas funcionalidades.

2.1.1 Elicitação de requisitos

Uma etapa crítica da Engenharia de Requisitos é a elicitação, sendo fundamental para identificar requisitos que orientem as escolhas técnicas e escolhas de valor para o cliente. Embora a elicitação seja a primeira atividade na engenharia, ela não acontece somente uma vez, pois é um processo iterativo (Belgamo; Martins, 2000) que visa garantir a funcionalidade e a adequação do produto ao seu propósito. A elicitação de requisitos (ou levantamento de requisitos) consiste em identificar as necessidades de um sistema por meio do diálogo com os

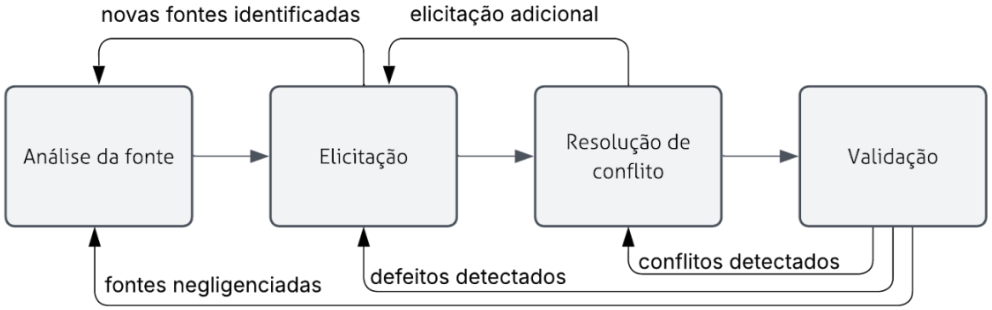
stakeholders, onde o levantamento deve ser focado em descobrir e compreender as necessidades dessas partes interessadas.

Segundo Glinz et al. (2010, p. 12), um *stakeholder*, ou parte interessada, é definido como uma "pessoa ou organização que influencia os requisitos de um sistema ou é impactada por ele". Dessa forma eles são as principais fontes para o levantamento dos requisitos e a não inclusão - ou inclusão tardia - de um *stakeholder* relevante pode gerar um impacto negativo na qualidade final do produto. O autor ainda acrescenta que

> os stakeholders podem ser encontrados em uma ampla área ao redor do sistema, desde usuários diretos e indiretos do sistema, gerentes (de negócio), pessoal de TI como desenvolvedores e operadores, até adversários e concorrentes, instituições governamentais e regulatórias, e muitos outros. (Glinz et al., 2010, p. 85)

Nesse sentido, o processo de elicitação envolve a interação com clientes e usuários finais para obtenção de informações sobre o domínio da aplicação, os serviços desejados, o desempenho esperado e outras restrições. Embora não haja um consenso único na literatura sobre suas atividades concretas, é estabelecido que a elicitação não é um evento isolado, mas sim uma atividade interconectada, um ciclo contínuo. A Figura 1 ilustra essa dinâmica, evidenciando que as etapas do processo de engenharia de requisitos podem ser realizadas em paralelo, em *loops*, ou sequencialmente, dependendo do contexto de cada situação (Glinz et al. 2010).

Figura 1 – Engenharia de Requisitos não é linear



Fonte: Adaptado de Glinz et. al (2010).

Na prática, a elicitação pode identificar novas fontes de informações, forçando um retorno à fase de análise. Da mesma forma que a validação funciona como um ponto de controle de qualidade: se um requisito estiver incorreto ou um conflito for detectado, o processo exige uma elicitação adicional para esclarecer ambiguidades. Reforçando a ideia de que não basta

“coletar” requisitos, é preciso garantir que eles estejam adequados antes de avançar para a fase de desenvolvimento.

O objetivo final desse esforço sempre será construir um sistema que resolva problemas para seus usuários e satisfaça as expectativas das partes interessadas (Glinz et al. 2010). É uma atividade complexa, pois envolve pessoas com diferentes conhecimentos e objetivos (Silva; Benitti, 2011) e é suscetível a falhas de comunicação que podem comprometer o projeto.

2.1.2 Técnicas de elicitação

Para estruturar o processo de descoberta das necessidades do *stakeholder* e melhorar a comunicação entre analistas e clientes, a engenharia de *software* estabeleceu um conjunto de abordagens metodológicas conhecidas como Técnicas de Elicitação de Requisitos. Glinz et al. (2010) afirmam que a seleção da abordagem mais adequada depende do contexto e consolida essas técnicas em grupos, como por exemplo as técnicas de Coleta e as técnicas de Desenho e Geração de Ideias. Para o escopo deste trabalho, duas técnicas são particularmente relevantes devido à sua aplicação direta no estudo, quais sejam: as entrevistas e o *brainstorming*.

As entrevistas são caracterizadas como técnica de coleta e, devido a sua flexibilidade, é provavelmente uma das técnicas de elicitação mais frequentemente utilizadas. Sua maior vantagem é poder ser usada para elicitar requisitos tanto abstratos quanto específicos e não necessitar de uma ferramenta própria. Normalmente, uma entrevista é uma sessão individual entre um engenheiro de requisitos (entrevistador) e um *stakeholder* (entrevistado), mas um pequeno grupo de entrevistados pode ser uma opção. Entretanto, sua maior desvantagem é a necessidade de ter objetivos bem definidos e uma boa preparação para obter resultados adequados, pois essa técnica pode consumir bastante tempo. Também é menos apropriada quando se deseja entrevistar um grande número de *stakeholders* (Glinz et al., 2010).

Já o *brainstormin*g é uma técnica de desenho e geração de ideias, caracterizada pela atuação em grupo. Segundo Pressman (2011), o *brainstorming* é colaborativo e favorece a geração rápida de ideias, sendo amplamente utilizada em contextos em que há restrições de tempo e necessidade de criatividade. Onde, um grupo de pessoas se reúne imaginando um cenário, então o assunto é discutido para elicitar os requisitos, o ponto crucial do *brainstorming* é o apoio ao desenvolvimento de novas ideias para uma questão ou problema, onde a quantidade prevalece sobre a qualidade e as ideias selecionadas servem como insumo para uma elicitação adicional, funcionando de forma excelente para grande volume de ideias. Entretanto, sua desvantagem é a falta de garantia de que essas ideias sejam viáveis ou relevantes, pois elas

precisam ser categorizadas, avaliadas e validadas posteriormente com outras abordagens (Glinz et al., 2010).

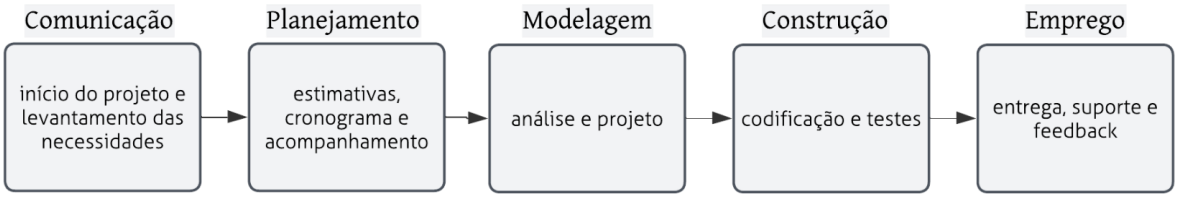
2.2 LINGUAGEM DE MODELAGEM UNIFICADA

A Linguagem de Modelagem Unificada (UML)[2] é uma linguagem visual utilizada para modelar sistemas computacionais, cujo objetivo é auxiliar os engenheiros de software a definirem características do sistema, tais como seus requisitos e comportamentos (Guedes, 2004). A modelagem de requisitos é um componente fundamental da ER e desempenha um papel importante na documentação de sistemas, oferecendo uma abordagem sistemática e disciplinada para a especificação e gerenciamento de requisitos, diferente da linguagem natural, que embora expressiva, pode sofrer de ambiguidade, omissões e inconsistências (Glinz et al., 2010).

2.3 PROCESSO DE SOFTWARE

Quando se trabalha na elaboração de um produto ou sistema, é importante seguir uma série de passos previsíveis ou um roteiro que ajude a criar um resultado de alta qualidade e dentro do prazo estabelecido. Tal roteiro é denominado "processo de *software*". Assim como a escolha da técnica de requisitos, a escolha do processo de *software* depende do contexto do projeto, tamanho da equipe e até mesmo a complexidade do sistema. A Figura 2, adaptada de Pressman (2011), ilustra os detalhes das fases típicas do desenvolvimento de sistemas, que são: comunicação, planejamento, modelagem, construção e emprego.

Figura 2 – Etapas típicas do processo de desenvolvimento de um software



Fonte: Adaptado de Pressman (2011).

[2] *Unified Modeling Language*.

2.4 HACKATHONS

Os *hackathons* são eventos que reúnem equipes com o objetivo de desenvolver soluções inovadoras para um problema proposto, geralmente relacionado à tecnologia, programação ou desenvolvimento de software. O termo surge da junção das palavras *hack* (explorar soluções criativas e técnicas) e *marathon* (maratona) e tem como objetivo desenvolver uma solução que cause impactos tanto internos (como no caso de uma empresa, por exemplo) quanto externos na forma de protótipos funcionais, importante definição para testar e validar ideias rapidamente. O primeiro hackathon que se tem notícia aconteceu no ano de 1999. Desde então, virou uma prática popular, especialmente nos Estados Unidos, mas que logo se espalharia por diversos países, inclusive no Brasil (Borges, 2019).

2.5 TECNOLOGIAS UTILIZADAS

2.5.1 Tecnologias de suporte

Para a diagramação do sistema, foi escolhido o *software* **Astah**, comumente utilizado por empresas e estudantes para a produção de diagramas de sistemas. O Astah é uma ferramenta de modelagem desenvolvida para criar e explorar designs, visualizar arquiteturas e comunicar-se de forma eficiente dentro das equipes e com as partes interessadas. Ela oferece suporte à criação de diagramas UML, fluxogramas, diagramas de fluxo de dados e diagramas de requisitos, proporcionando uma solução abrangente para desenvolvedores, engenheiros e arquitetos técnicos (Changevision, 2025).

2.5.2 Tecnologias de desenvolvimento

O *Java*, com seu ecossistema maduro e vasta disponibilidade de bibliotecas, foi selecionado para o desenvolvimento do *backend* e da *API* (Interface de Programação de Aplicações) para gerenciar os agendamentos e dados dos usuários. Para o *frontend*, optou-se pelo *Angular*, um *framework* conhecido por sua estrutura opinativa que pode acelerar o desenvolvimento de interfaces complexas e ricas em funcionalidades, facilitando a criação da plataforma de agendamento acessível via navegador. No que se refere ao aplicativo móvel, a adoção do *Flutter* permitiu o desenvolvimento ágil de uma interface nativa e performática para *Android* e *iOS,* a partir de uma única base de código, otimizando o esforço da equipe.

3 METODOLOGIA

A metodologia para compreender o processo de marcação de consultas baseou-se em uma abordagem qualitativa e foi adaptada a um cenário de tempo limitado, combinando análise documental, *brainstorming* e entrevistas informais. Inicialmente, foi realizada uma etapa de levantamento em equipe para entender o fluxo do processo baseado no conhecimento implícito, seguida pela condução de entrevistas com os usuários do sistema para validar esse entendimento.

Os dados coletados com os usuários permitiram a elaboração de um fluxograma (Figura 3), cuja análise será apresentada na seção 3.2.2, que ilustra as etapas da marcação. Adicionalmente, uma entrevista foi conduzida com uma médica para obter uma perspectiva profissional do processo, validando e enriquecendo as informações obtidas com os usuários.

3.1 O HACKATHON DA LOGAP

Em dezembro de 2023 ocorreu o “Hackathon da Logap”, um evento muito semelhante a uma “maratona de programação”, cujo objetivo foi promover uma competição entre equipes para o desenvolvimento de uma solução em um curto intervalo de tempo. A *LogAp Sistemas*, empresa especializada no desenvolvimento de soluções digitais personalizadas sob medida, organizou o *hackathon* como parte das comemorações de seu décimo aniversário, direcionando a participação exclusivamente para seus colaboradores.

Os participantes tiveram como meta desenvolver uma aplicação com base em um tema previamente definido: “**Desenvolver soluções que ampliem o acesso da população a serviços essenciais de utilidade pública**”. A formação das equipes ocorreu por meio de um sorteio, e os participantes tiveram um prazo de quase dois dias para conceber e implementar a solução. Além disso, como requisito obrigatório, cada grupo deveria contar com, no mínimo, um Analista de Requisitos. Como resultado, as equipes vencedoras levaram prêmios de diversas categorias e a equipe do MarcaSUS foi a grande campeã da primeira edição do evento.

3.2 COLETA DE DADOS

3.2.1 Entendimento do processo de marcação de consultas

A compreensão do processo de marcação e das necessidades dos usuários constitui uma etapa fundamental que precede a especificação formal dos requisitos de um software (Silva; Benitti, 2011). Nesse contexto, para especificar em um tempo curto e com pressão, baseado em

seu conhecimento implícito e da equipe como usuários do sistema público, a autora dedicou um tempo para escrita de informações e entrevistas com usuários conhecidos, a fim de embasar o desenvolvimento do sistema.

De acordo com a análise documental realizada na data do evento nos principais noticiários da cidade, a Pesquisa Nacional de Saúde - PNS (IBGE, 2019) relata que cerca de 70% dos brasileiros que buscam atendimento médico recorrem à rede pública. As Unidades Básicas de Saúde (UBS) representam um papel central nesse cenário, sendo mencionadas por 54,1% dos entrevistados na região Nordeste como sendo o principal ponto de acesso aos serviços de saúde. Segundo Cunha (2020, s.p.), "pacientes que precisam agendar exames laboratoriais e/ou marcar consulta médica passaram a dormir na unidade para conseguirem uma ficha de atendimento". Outro obstáculo enfrentado pelos usuários é a insegurança, onde em algumas regiões, como na Grande Natal, pessoas na fila da vacinação têm sido alvo de assaltos (Teixeira, 2021), evidenciando a urgência de alternativas mais seguras e organizadas para o acesso à saúde pública.

Tais fatos ressaltam a sobrecarga do sistema e a necessidade de melhorias no processo de agendamento, evidenciando para a autora e sua equipe, a relevância do MarcaSUS para ampliar o acesso da população a serviços essenciais de utilidade pública. Ademais, essa realidade revela alguns dos problemas que comprometem a universalidade do acesso ao Sistema Único de Saúde (SUS), os quais vêm causando transtornos aos cidadãos, especialmente os mais vulneráveis.

Partindo desse entendimento, a compreensão do processo se deu principalmente pelas experiências pessoais, que serão detalhadas a seguir, e por entrevistas informais realizadas com pessoas próximas aos participantes.

Como mencionado previamente, os requisitos de um sistema correspondem às descrições das funcionalidades e comportamentos esperados. No caso do desenvolvimento do MarcaSUS, o levantamento dos requisitos não seguiu fases iterativas[3], como normalmente ocorre em projetos com maior disponibilidade de tempo e interação com *stakeholders*. Diante do desafio apresentado e da restrição de tempo, a equipe não pôde seguir um processo formal de elicitação, em vez disso, adotou-se uma abordagem pragmática e ágil, combinando duas técnicas principais: entrevistas informais e o *brainstorming*.

[3] Na qual ocorre a retroalimentação das atividades (Fernandes, 2017), ou seja, pode ocorrer elicitação em todas as outras fases do processo.

a. ***Brainstorming***: a equipe promoveu uma sessão colaborativa de ideação para discutir e priorizar um conjunto inicial de requisitos, aproveitando o conhecimento prévio e a vivência de membros da equipe como usuários do SUS.
b. **Entrevistas (informais):** foram conduzidas conversas diretas com usuários do sistema público e um profissional da área da saúde, para uma rápida compreensão de suas principais necessidades, pontos de insatisfação e validação dos insumos do *brainstorming*.
c. **Análise Documental**: realizou-se uma pesquisa em reportagens e conteúdos da mídia para contextualizar a dimensão do problema do agendamento de consultas no SUS e validar a importância do protótipo.

As entrevistas, realizadas com conhecidos que são usuários do SUS, serviram para validar as "dores" iniciais. Contudo, foi a sessão de *brainstorming* entre os membros da equipe que efetivamente definiu o escopo do projeto. A partir da discussão colaborativa, foi gerado um conjunto mínimo de requisitos funcionais considerado viável para o tempo disponível para um Produto Mínimo Viável (MVP), conforme apresentado na Tabela 1. O resultado dessas definições foi materializado em anotações, que guiaram o desenvolvimento das telas principais do sistema.

Tabela 1 – Requisitos funcionais do sistema

ID	Ator(es)	Requisito Funcional	Descrição
RF01	Paciente e Funcionário	Marcação de Consulta	Agendar consultas médicas.
RF02	Paciente e Funcionário	Desmarcar Consulta	Cancelar consultas previamente agendadas.
RF03	Paciente e Funcionário	Visualização de Consultas Agendadas	Exibir consultas confirmadas, informando data e hora.
RF04	Funcionário	Preenchimento de Agenda	Adicionar e remover horários disponíveis na agenda.
RF05	Funcionário	Visualização de Agenda	Visualizar agenda preenchida/não preenchida.

Fonte: Elaboração própria (2025)

3.2.2 Resultado das entrevistas com os sujeitos

Para a segunda forma de coleta de dados, foram conduzidas entrevistas com duas usuárias do sistema público de saúde que se identificam como mulheres, de faixas etárias

distintas (25-29 anos e 60-64 anos). Ambas as entrevistas foram informais e realizadas através do aplicativo de comunicação *Whatsapp*. Inicialmente, a conversa foi norteada pelas perguntas:

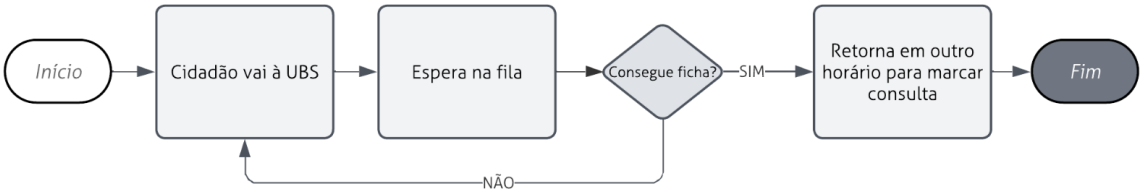
1) Como funciona a marcação de consultas?

2) Tem ficha para todo mundo ou é por ordem de chegada?

3) Existe alguma forma de marcar ou visualizar consultas em algum aplicativo?

Em resposta à terceira pergunta, as entrevistadas prontamente informaram desconhecer, mencionando apenas aplicativos relacionados à vacinação e ao Cartão Nacional de Saúde, garantindo a execução do princípio 8 da Engenharia de Requisitos: Inovação. As respostas às demais questões revelaram o ponto central de insatisfação, corroborado por ambas as entrevistadas. A Figura 3 exemplifica o processo de agendamento presencial, caracterizado como manual e ineficiente, pois, ainda é realizado por meio da entrega presencial de fichas, geralmente entre 7h e 8h da manhã, onde aproximadamente 15 a 20 fichas são distribuídas diariamente.

Figura 3 – Fluxo da jornada do paciente no agendamento de consultas



Fonte: Elaboração própria (2025).

Essa dinâmica, que reafirma que a oferta não atende à demanda, impõe aos pacientes a necessidade de chegar às unidades com antecedência e a obrigação de retornar posteriormente para concluir a marcação da consulta. O segundo ponto de queixa em comum, refere-se à desmarcação dessas consultas, que ainda é ineficaz, pois também demanda a presença física do paciente na unidade para notificar sua ausência.

3.2.3 Resultado da entrevista com médica de uma UBS

Para finalizar e alinhar o entendimento da realidade das UBSs, foi realizada uma entrevista com uma médica que já atuou em uma UBS da cidade de Natal. Ela destacou que um dos principais problemas enfrentados no atendimento é a ausência dos pacientes que agendam consultas, mas que não comparecem e nem comunicam a ausência. Segundo a profissional, essa situação gera ociosidade na agenda médica, uma vez que, mesmo sem atendimento, o médico permanece na unidade cumprindo o horário de trabalho. Além disso, relatou que todo o processo

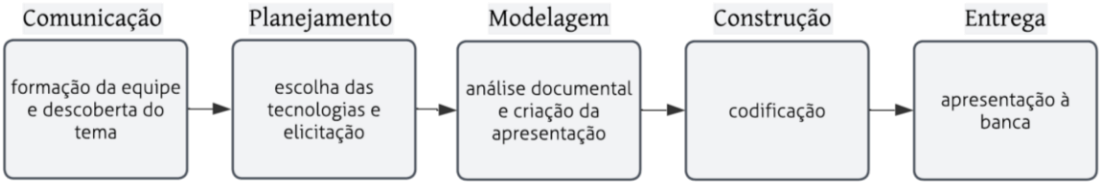
ser realizado manualmente, sem qualquer tipo de digitalização, tornava a gestão mais burocrática e suscetível a falhas. Por fim, ainda citou que é comum a venda de fichas, evidenciando a ineficiência do processo, principalmente quando se trata do acesso igualitário.

3.3 PROCESSO DE DESENVOLVIMENTO ADOTADO

Uma vez que este trabalho foi produzido no contexto de um *hackathon*, não foi viável a adoção integral de um processo formal de desenvolvimento de *software*. Em vez disso, foi adotada uma abordagem focada na entrega de um projeto funcional. Assim que a equipe teve conhecimento do tema da competição, houve uma reunião para gerar e discutir o que cada um tinha de informação, dúvida ou ideia inicial acerca do processo de marcação de consultas. A escolha se deu não apenas por sua agilidade, mas por ser a abordagem mais tangível e prática que a equipe possuía.

A Figura 4 detalha como o processo do desenvolvimento foi adaptado, iniciado pela formação das equipes, a descoberta do tema e seguido do entendimento do que seria o sistema (sessão de *brainstorming*). A escolha das tecnologias foi feita com base na familiaridade e na experiência prévia dos participantes, uma vez que os requisitos mínimos foram estabelecidos e a competição formalmente iniciada, os participantes partiram para a fase de desenvolvimento.

Figura 4 – Processo de software adaptado pela equipe



Fonte: Elaboração própria (2025).

Durante a codificação, a prática mais próxima de um processo de *software* foi a programação em par (*pair programming*) e a participação das monitorias oferecidas pela organização, que serviram como ponto de orientação. É importante frisar que a definição dos requisitos não foi documentada formalmente, apenas em forma de *slide* para a apresentação à banca no final da competição.

Mesmo de maneira adaptada, este fluxo simplificado de trabalho reflete as atividades de um processo de software citadas por Sommerville (2011): especificação, projeto e implementação e validação, sucessivamente adaptando-as para *brainstorming*, fase de codificação e mentorias.

4 RESULTADOS

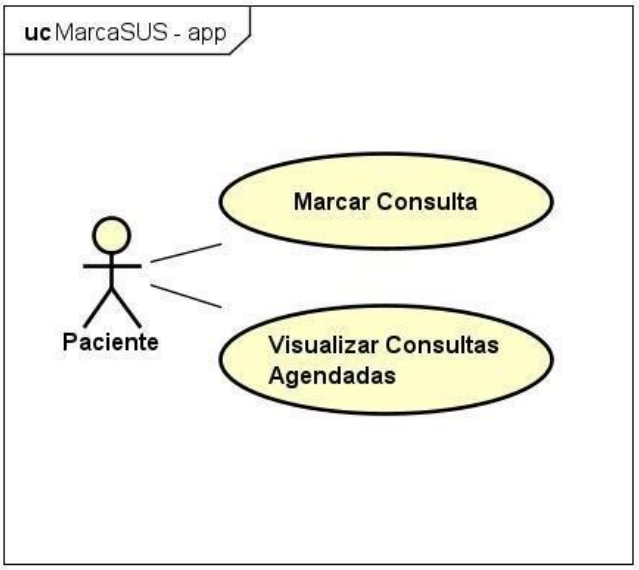
Neste capítulo serão apresentados os resultados obtidos por meio do estudo de caso do desenvolvimento do protótipo funcional do MarcaSUS e a discussão dos resultados desse processo.

4.1 MARCASUS

4.1.1 Casos de Uso

Os diagramas de casos de uso representados nas Figuras 5 e 6 foram elaborados para a execução deste trabalho e ilustram as funcionalidades efetivamente implementadas durante o desenvolvimento do sistema no *hackathon*. A Figura 5 apresenta o caso de uso do perfil "Paciente", evidenciando as funcionalidades de Marcar Consulta e Visualizar Consultas Agendadas, que foram priorizadas pela equipe considerando o fluxo essencial para os usuários finais.

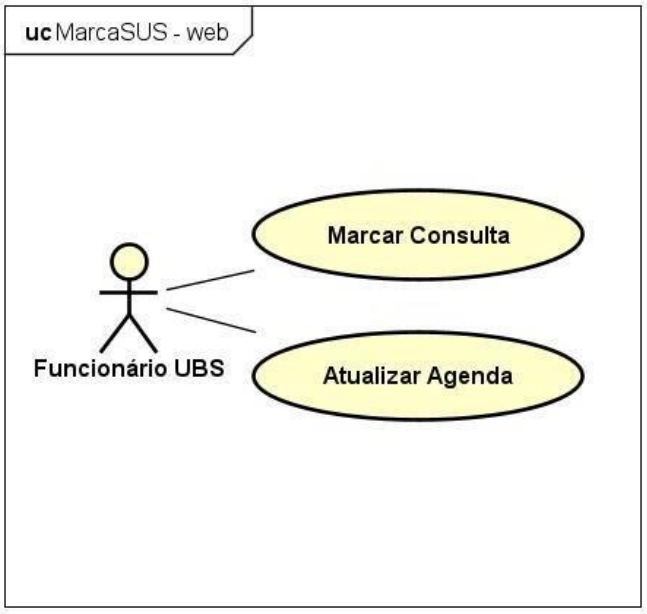
Figura 5 – Diagrama de caso de uso do paciente



Fonte: Elaboração própria (2025).

Por sua vez, a Figura 6 demonstra os casos de uso correspondentes ao "Funcionário da UBS", o qual possui acesso à plataforma *web* e pode realizar ações como Atualizar Agenda e Marcar Consulta.

Figura 6 – Diagrama de caso de uso do funcionário



Fonte: Elaboração própria (2025).

4.1.2 Plataforma *Web*

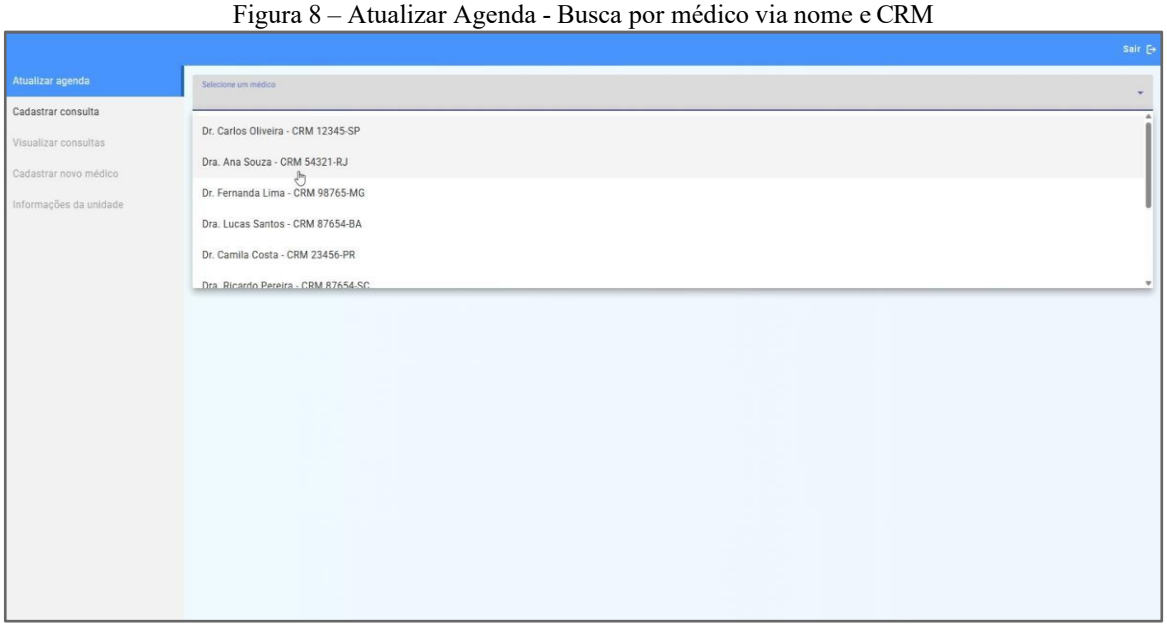
Como mencionado previamente, a plataforma web, de uso exclusivo do funcionário da UBS, foi implementada com diversas restrições de funcionalidades, contendo apenas as telas de Atualizar Agenda do médico e Marcar Consulta. Inicialmente o usuário faz o *login* (Figura 7), observa-se que a aplicação realiza apenas a validação do CPF, uma vez que não foi possível implementar, nesta versão, o mecanismo completo de autenticação com senha.

Figura 7 – Tela de login da plataforma web com validação de CPF

Fonte: Elaboração própria (2025).

Em seguida, realiza a busca pelo médico através do nome e CRM por meio de uma caixa de busca para seguir com o processo de atualização da agenda médica (Figura 8).

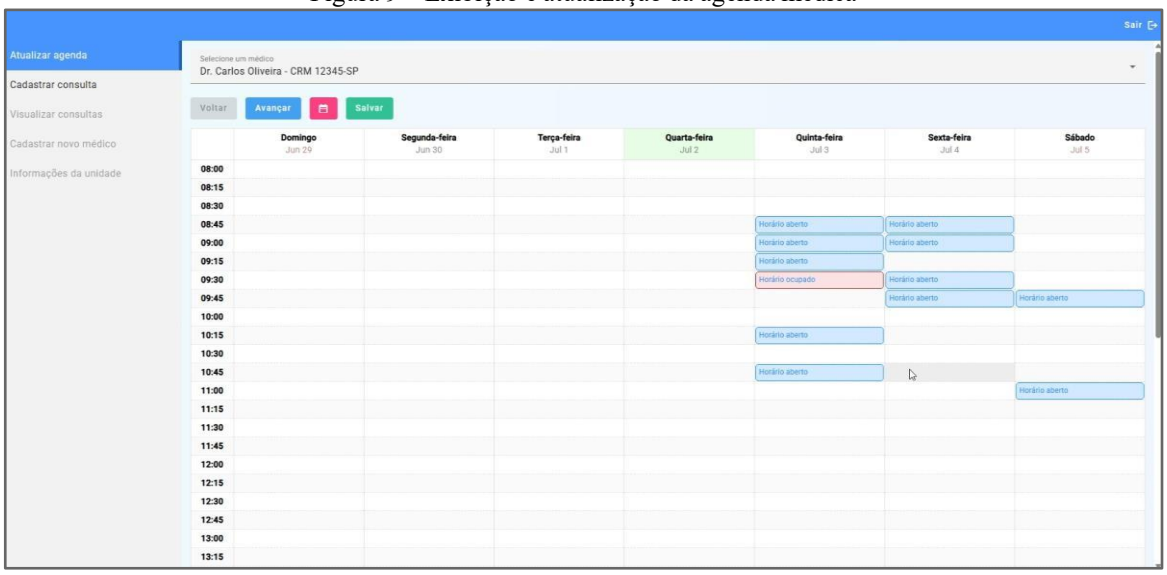
Figura 8 – Atualizar Agenda - Busca por médico via nome e CRM



Fonte: Elaboração própria (2025).

Ao localizar o profissional, sua agenda é exibida na tela, a aplicação permite que o funcionário marque horários disponíveis, indicando novos períodos para agendamento de consultas (Figura 9).

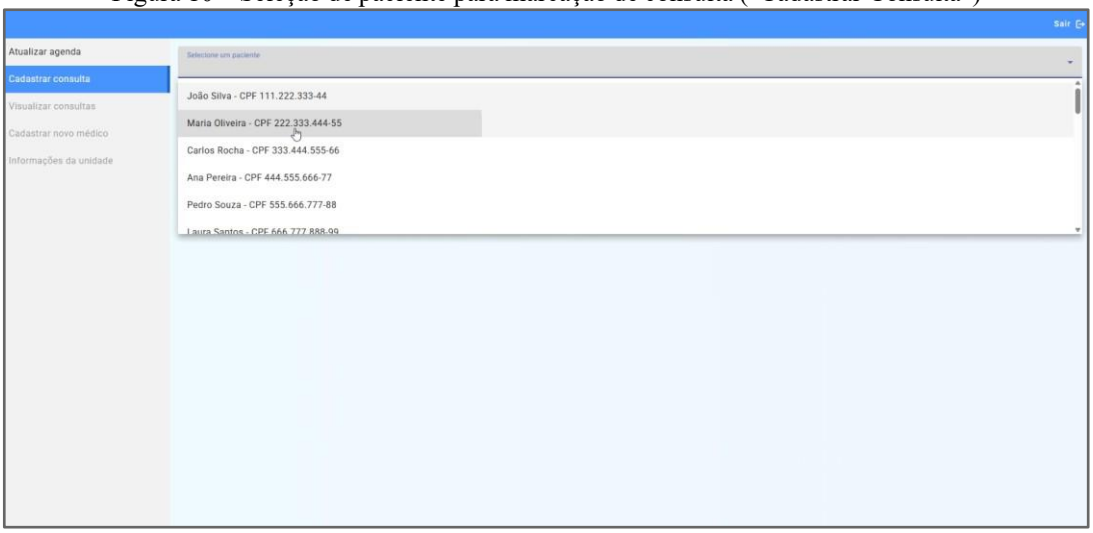
Figura 9 – Exibição e atualização da agenda médica



Fonte: Elaboração própria (2025).

Na Figura 10, o funcionário seleciona o paciente pelo campo “Nome + CPF” para iniciar o agendamento da consulta, a funcionalidade está rotulada como Cadastrar Consulta.

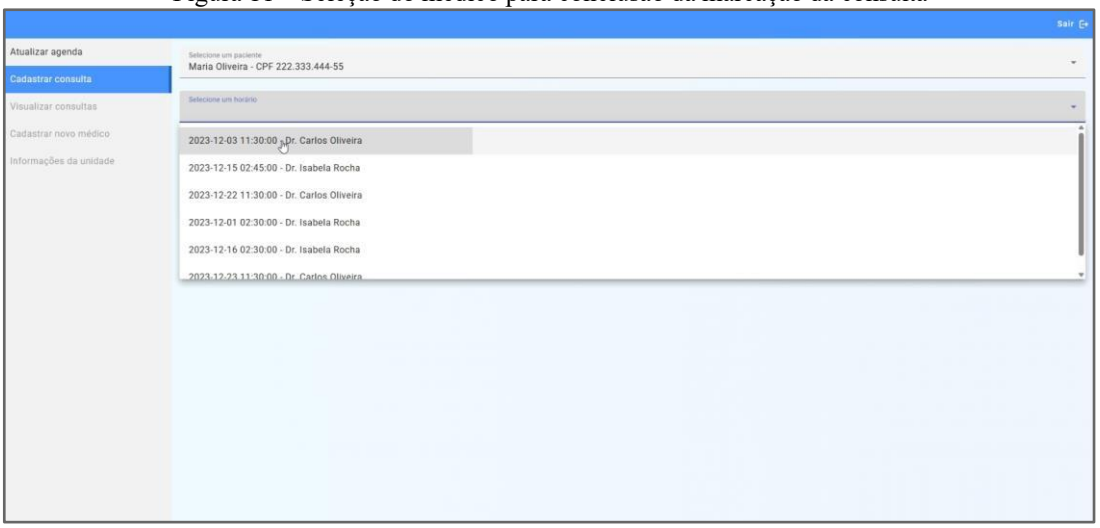
Figura 10 – Seleção de paciente para marcação de consulta (“Cadastrar Consulta”)



Fonte: Elaboração própria (2025).

Por fim, a Figura 11 mostra a continuação do processo, em que o profissional da UBS escolhe o médico correspondente, utilizando novamente os campos “Nome + CRM”, para concluir o agendamento da consulta do paciente selecionado na etapa anterior.

Figura 11 – Seleção do médico para conclusão da marcação da consulta



Fonte: Elaboração própria (2025).

4.2 ESTUDO DE CASO - DESAFIOS DA ELICITAÇÃO NO *HACKATHON*

Esta seção analisa os desafios da elicitação no evento, os resultados apresentados, comparando a prática observada no *hackathon* com o cotidiano da autora e a teoria da Engenharia de Requisitos.

4.2.1 Análise da elicitação de requisitos no cotidiano da autora

Como analista de sistemas em formação pelo IFRN, com a disciplina de engenharia de requisitos já concluída e atuando profissionalmente na área de análise de requisitos, em um cenário natural do dia a dia da autora, a elicitação é conduzida de forma mais estruturada, envolvendo diferentes técnicas (como entrevistas, análise de documentos, reuniões colaborativas) e a participação ativa de *stakeholders*. No exercício de sua função, o trabalho diário da autora é atuar como uma ponte entre as necessidades do cliente e a equipe de desenvolvimento. Em um cenário ideal, o processo de elicitação de requisitos não é apenas uma "coleta" de pedidos, mas uma fase colaborativa de descoberta, negociação e validação.

No processo de recepção de uma nova aplicação, os clientes da empresa na qual a autora trabalha, utilizam o método *Lean Inception* para auxiliar no entendimento do *software* a ser desenvolvido. Segundo Caroli (2018), o método é eficaz para decidir o MVP e funciona como um *workshop* dividido em várias etapas e atividades que irão direcionar a equipe na construção do produto ideal. Uma vez que a *Lean Inception* é encerrada, é iniciado o processo de "refinamento", caracterizado por reuniões que reúnem os *stakeholders*, todos alinhados e com um único objetivo: compreender o que foi levantado, validar as análises feitas previamente e revisar os requisitos, a fim de deixar o *backlog* do sistema preparado.

O processo atual de elicitação de requisitos, na prática diária do ambiente de trabalho da autora, em fases, será apresentado a seguir:

a. **Fase 1:** Preparação e Planejamento

   Antes de qualquer conversa com um *stakeholder*, o trabalho começa com uma imersão no contexto, onde, às vezes, temos acesso livre ao plano de negócios e aos objetivos estratégicos que motivaram o projeto. O primeiro passo é entender o "porquê" por trás do *software*, isso permite à equipe da autora a fazer perguntas mais direcionadas e guiar as discussões para além de funcionalidades superficiais;

b. **Fase 2**: Mapeamento de *Stakeholders*

   Com o apoio da gestão do projeto por parte do cliente, todas as partes interessadas já são identificadas: usuários finais de diferentes perfis, gestores do projeto, equipes de suporte e testes, desenvolvedores e especialistas do sistema;

c. **Fase 3**: Definição do Escopo

   A equipe está alinhada no que diz respeito a uma visão inicial e consensual sobre o projeto, o time já sabe o que o sistema deve fazer e, igualmente importante, o que ele não deve fazer;

d. **Fase 4:** Análise e Documentação

   Aqui, todo o material coletado (notas, gravações, *prints*) é processado após as sessões de refinamento, as informações são organizadas e às vezes classificadas em requisitos funcionais, não funcionais ou regras de negócio. As necessidades são traduzidas em artefatos que serão utilizados no desenvolvimento, como Histórias de Usuário (*User Story*), Critérios de Aceitação ou Protótipos, caso a equipe julgue necessário.

Diferente de um cenário de *hackathon*, as reuniões de refinamento acontecem pelo menos duas vezes por semana, reúnem as partes interessadas no produto e ocorrem paralelamente ao desenvolvimento iniciado na primeira *sprint*. Em síntese, a participação em um evento caracterizado por restrições de tempo e de *stakeholders*, evidenciou para a autora que a elicitação de requisitos em um cenário ideal é um processo proativo, estruturado e profundamente colaborativo, impossibilitado de ocorrer em um cenário com prazos curtos e de competição.

4.2.2 Análise dos principais desafios da elicitação no *hackathon*

Diferentemente do ambiente corporativo tradicional, onde observamos que a análise de requisitos é realizada de forma estruturada, com o suporte de métodos e metodologias, reuniões regulares, acesso aos *stakeholders* e a então definição de critérios de aceitação e histórias de usuários, a dinâmica de um *hackathon* impõe um cenário complexo e muito mais desafiador.

Os principais desafios enfrentados foram a ausência do usuário final e a redução do tempo, os quais impediram a execução de um ciclo de validação, uma etapa considerada fundamental por autores como Pfleeger (2004). Para contornar essa limitação, a equipe utilizou duas estratégias principais:

a. **A equipe como usuário final**: os membros atuaram como representantes dos usuários, usando suas próprias experiências para tomar decisões de design e funcionalidade.
b. **Foco no MVP:** a priorização foi seletiva, focando exclusivamente no fluxo essencial do agendamento para garantir a entrega de um protótipo funcional, em detrimento de funcionalidades secundárias.

Além dos desafios principais citados acima, fica evidente que a pressão de desenvolver um protótipo funcional e a não utilização de metodologias ágeis formais foram desafios secundários que exigiram que a elicitação ocorresse de forma rápida, baseada em entrevistas por mensagens e experiência da equipe. A ausência do cenário em que comumente estamos inseridos no processo de desenvolvimento de uma aplicação, tornou o processo de levantamento

de requisitos um exercício desafiador, por vezes inseguro e que necessitou de adaptação e tomada de decisão ágil, o foco não foi mais um sistema funcional voltado às necessidades do cliente, e sim em entregar o máximo possível da solução dentro do prazo estipulado, mesmo diante de incertezas, o que fere o conceito de entrega de valor da ER.

Diante dessa perspectiva, se torna praticamente impossível realizar qualquer cerimônia como *Lean Inception,* seguir um processo de *software* ou até mesmo validar requisitos com as partes interessadas. O máximo que a autora obteve foi a lista de quais funcionalidades a aplicação poderia ter para sanar os principais problemas levantados, para que então o processo de desenvolvimento fosse o mais objetivo possível e pudesse contemplar o máximo de funcionalidades em quase 36 horas.

Dentro do cenário da competição, há sempre muita brincadeira, distrações e vontade de fazer mais do que pode. Em um dos alinhamentos com a mentoria da competição, a autora demonstrou o interesse em criar uma documentação do sistema e um diagrama de Casos de Uso. Porém, recebeu a orientação da mentoria de que “é fundamental no dia a dia, mas esse não é o momento para isso”, evidenciando que a engenharia de requisitos é fundamental na construção de sistemas funcionais, porém não onde o resultado esperado é apenas um protótipo funcional entregue como MVP.

4.2.3 Análise das técnicas adotadas

A escolha das técnicas foi claramente influenciada pelas restrições do ambiente. Enquanto a literatura (Sommerville, 2011) aponta as entrevistas como uma técnica rica para aprofundamento, no contexto do estudo elas ocorreram de maneira informal e serviram apenas como uma validação inicial.

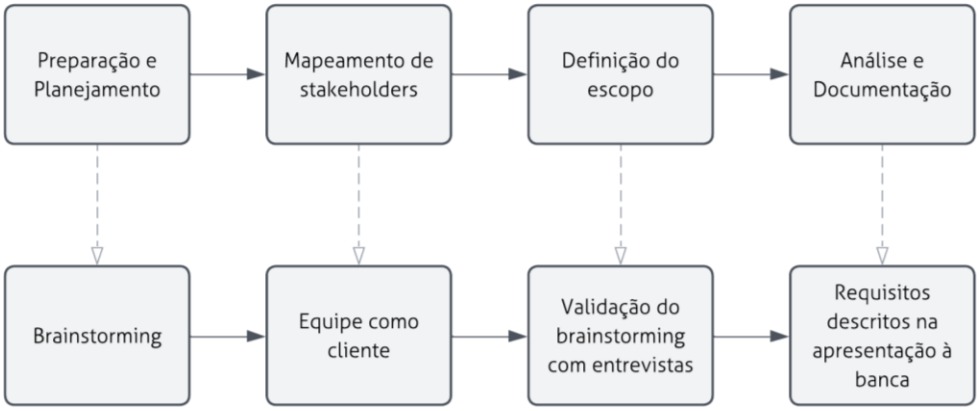
O *brainstorming*, por sua vez, emergiu como a ferramenta central. Sua natureza ágil, colaborativa e de baixo custo se mostrou perfeitamente alinhada às necessidades do *hackathon*. A técnica permitiu que o conhecimento prévio e a experiência pessoal dos próprios membros da equipe fossem externalizados e transformados em requisitos funcionais, substituindo a figura do *stakeholder* formal. Na ausência de acesso a essa figura, o conhecimento implícito dos próprios membros foi o recurso mais valioso e acessível para dar partida ao projeto.

4.2.4 Análise da comparação do Processo Ideal vs. Processo Adaptado

Para sintetizar a análise, os artefatos a seguir comparam as atividades do processo de elicitação “ideal” baseado no dia a dia autora, com a forma como foram adaptadas ou suprimidas durante o *hackathon*. As figuras contribuem para a compreensão das diferenças entre o processo de elicitação de requisitos ideal e o que foi efetivamente praticado durante o *hackathon*.

A Figura 12 representa o fluxo de atividades citado previamente, que compõem o processo de elicitação no cotidiano da autora, incluindo as etapas de preparação e planejamento, mapeamento de *stakeholders*, definição do escopo, análise e documentação, realizando uma conexão das etapas adotadas pela equipe no contexto específico do MarcaSUS, partindo da sessão de *brainstorming*, substituindo o mapeamento dos *stakeholders* pelas suposições da equipe como cliente e definindo o escopo através da validação informal por meio das entrevistas, até a documentação dos requisitos realizada na forma da apresentação final.

Figura 12 – Elicitação no cotidiano *versus* elicitação no *hackathon*



Fonte: Elaboração própria (2025).

Por sua vez, a Tabela 2 demonstra que, embora o processo tenha sido simplificado, duas tarefas importantes da elicitação (entender problema e definir escopo) estiveram presentes, ainda que de forma adaptada.

Tabela 2 – Comparativo entre o Processo de Elicitação Ideal e o Praticado no *Hackathon*

Atividade "ideal"	**Prática adotada no hackathon**	**Implicações**
Preparação e planejamento	Não houve, a escolha da técnica foi mais importante.	Risco de não escolher a melhor técnica e gerar retrabalho ao precisar refatorar o sistema, uma vez que os requisitos podem não ter sido levantados da melhor forma.
Mapeamento de *stakeholders*	Foco no resultado da elicitação, não há *stakeholders* estruturados.	Obtenção rápida de um escopo mínimo, mas com baixa profundidade, podendo resultar em um produto que não resolve um problema real.
Validação com *stakeholders*	Substituída pela validação interna da própria equipe.	Risco de o produto não atender às reais necessidades do usuário final.
Documentação formal de especificação	Inexistente, substituída por uma lista informal de requisitos.	Requisitos ambíguos ou incompletos, falta de rastreabilidade, podendo gerar gastos na evolução dos requisitos.

Fonte: Elaboração própria (2025).

É inegável que essas adaptações elevaram o risco de desalinhamento com as reais necessidades do usuário. Atuar como cliente, escolher uma técnica inadequada e, principalmente, a ausência de um ciclo de validação com as partes interessadas são práticas que podem resultar em requisitos baseados em suposições, no desenvolvimento de funcionalidades irrelevantes e na necessidade de refatoração em um projeto real. Contudo, essa troca consciente entre a segurança do processo e a velocidade da entrega é, precisamente, o que define a essência de um *hackathon*. A experiência do MarcaSUS corrobora a visão de que, a Engenharia de Requisitos, não deve ser vista como um conjunto rígido de regras, mas sim como um guia metodológico flexível, cujas práticas podem e devem ser adaptadas de acordo com as necessidades e os objetivos de cada projeto.

4.2.5 Benefícios da Engenharia de Requisitos (ER)

Mencionamos anteriormente que a aplicação da ER traz uma série de ganhos para o processo de desenvolvimento de sistemas e ultrapassa uma simples documentação, representando um pilar que gera benefícios para a organização. O desenvolvimento de qualquer sistema é custoso e envolve riscos significativos para todos os participantes (Glinz et al., 2010), naturalmente requisitos evoluem ao longo do desenvolvimento de uma aplicação e um dos maiores desafios é rastrear a evolução natural dele. Em outras palavras, Fagundes (2011) diz

que confiar na memória da equipe para rastrear o impacto de uma mudança ou de uma nova funcionalidade não é sustentável.

Uma Engenharia de Requisitos (ER) bem executada agrega valor ao sistema e minimiza os riscos de falhas ou modificações custosas nas etapas posteriores ao desenvolvimento, Glinz et al. (2010) destacam alguns benefícios, diretos e indiretos:

- Fornece uma base apropriada para estimar o esforço e o custo do desenvolvimento.
- Aprimora a qualidade do software, o que se reflete em menos retrabalho e melhor atendimento às expectativas dos clientes.
- Cria, estimula e assegura o entendimento compartilhado entre stakeholders, Engenheiros de Requisitos e desenvolvedores.
- É pré-requisito para testar os sistemas adequadamente.
- Minimiza o risco de falhas ou modificações caras em etapas posteriores do desenvolvimento.

Em resumo, o valor econômico da Engenharia de Requisitos é, em sua maioria, indireto, pois o benefício principal muitas vezes reside na redução dos custos durante a implementação e operação, e não em um retorno financeiro direto pela atividade em si (Glinz et al. 2010). Assim, a ER é um investimento em qualidade que previne problemas caros no futuro (Pressman, 2011) e, para o *hackathon*, é quase impossível medir o prejuízo que a realização da forma simplificada da ER trará ao sistema.

4.2.6 Análise da aplicação dos princípios da ER no cenário do *hackathon*

Para sintetizar a definição dos 9 princípios da Engenharia de Requisitos abordados no item 2.1 deste trabalho junto ao protótipo do MarcaSUS, a Tabela 3 oferece um panorama consolidado com o contraste de cada princípio e sua aplicação prática dentro do *hackathon*, evidenciando as adaptações e suas implicações no projeto.

Tabela 3 – Análise Comparativa dos Princípios da ER no Contexto do Hackathon

Princípio	**Aplicação no *hackathon***	**Adaptação - Implicação**
1. Orientação para o valor	Foco no impacto para a banca avaliadora.	Valor de curto prazo priorizado acima do longo prazo e da qualidade do produto.
2. Stakeholder	A equipe foi o próprio *stakeholder*.	Mais ágil, mas com risco de desalinhamento maior.
3. Entendimento comum	Conhecimento implícito da equipe.	Rápido e orgânico, porém enviesado pela visão da equipe.
4. Contexto	Incompleto, poucas opiniões para levar em consideração.	Suficiente para o MVP, mas com risco de ignorar o ambiente real.
5. Problema - requisito - solução	Elo frágil entre problema real e requisito.	O protótipo resolve o problema entendido pela equipe, não necessariamente o problema real.
6. Validação	Mínima e sem usuário final.	Trocando segurança pela velocidade da implementação.
7. Evolução	Inexistente, foco no MVP.	Adequado para a competição, inviável para um produto real.
8. Inovação	Atendido, solução nova para o contexto.	Ponto mais forte do projeto, pode gerar um diferencial competitivo.
9. Trabalho disciplinado	Atendido com foco no escopo do MVP, não no processo.	Permitiu a entrega de pelo menos duas telas no prazo estipulado.

Fonte: Elaboração própria (2025).

A análise da aplicação dos nove princípios da Engenharia de Requisitos no contexto do *hackathon* revela que o ambiente de alta restrição forçou uma aplicação seletiva e, por vezes, a violação deliberada de algumas de suas diretrizes fundamentais.

O princípio de Orientação para o valor, por exemplo, não foi cumprido, mas foi adaptado: em vez de focar no valor de negócio a longo prazo, a equipe direcionou seus esforços para entregar um MVP capaz de impressionar a banca avaliadora, ainda que isso significasse não aprofundar na qualidade final do serviço ao usuário. Da mesma forma, o princípio dos *Stakeholders* foi contornado quando a equipe assumiu o papel do próprio cliente, uma estratégia ágil, porém arriscada, que ignora a premissa de que o sucesso do *software* depende de atender às expectativas das partes interessadas reais.

Consequentemente, o princípio da Validação foi o mais sacrificado, a necessidade de confirmar os requisitos com os usuários finais, uma etapa crucial para garantir o alinhamento, foi preterida em favor da velocidade de implementação. De forma similar, o princípio da

Evolução foi ignorado, uma vez que a ausência de uma documentação formal tornaria o gerenciamento de qualquer evolução futura extremamente complexo. A teoria reconhece que é natural um requisito evoluir e por isso é crucial gerenciar essa evolução para manter os requisitos estáveis, pois o custo da mudança pode ser alto. Em contrapartida, o princípio que mais foi contemplado foi o da Inovação, uma vez que as pesquisas e entrevistas confirmaram que a solução proposta era inédita para o contexto local, tornando-se o principal êxito do projeto.

5 CONSIDERAÇÕES FINAIS

O presente trabalho se propôs a analisar as adaptações e os desafios do processo de elicitação de requisitos em um ambiente de desenvolvimento com restrições, como o de um *hackathon*, utilizando como estudo de caso o desenvolvimento do protótipo MarcaSUS. A pesquisa investigou as estratégias adotadas pela equipe diante da impossibilidade de aplicar um ciclo formal da Engenharia de Requisitos, com destaque para o uso de *brainstorming* e entrevistas informais. O principal resultado emergiu da análise e comparação entre o processo de elicitação ideal, vivenciado no cotidiano profissional, e o processo adaptado durante o evento.

Essa pesquisa revelou que, embora a abordagem adaptada tenha sido essencial para a entrega de um protótipo funcional, ela exigiu a priorização da implementação de um MVP, sacrificando alguns princípios fundamentais da ER, como a Validação com o usuário final e a participação formal de *Stakeholders*. Em contrapartida, o princípio da Inovação foi plenamente atendido. Considerando a priorização da velocidade, contudo, houve como consequência direta um escopo de implementação seletivo, onde o desenvolvimento de diversas telas secundárias foi comprometido para garantir a entrega do fluxo principal dentro do prazo estipulado, que reflete a natureza de um *hackathon*.

Conclui-se, portanto, que a Engenharia de Requisitos não é um conjunto rígido de regras, mas um guia flexível. A adaptação de suas práticas pode ser benéfica em ambientes de alta restrição para alcançar objetivos de curto prazo. Todavia, essa agilidade não substitui a necessidade de uma elicitação robusta, validada e aprofundada, que é indispensável para o desenvolvimento de um *software* sustentável e verdadeiramente alinhado às necessidades do cliente a longo prazo.

No que se refere ao processo do levantamento de requisitos, diante da impossibilidade de executar um ciclo completo de ER e da combinação adaptada das técnicas de entrevista e *brainstorming*, nesse sentido, a autora sugere que o *brainstorming* possa ser a técnica que mais se adequa para *hackathons*, pois sua natureza participativa e de baixo custo operacional permite levantar rapidamente os principais requisitos com base na colaboração entre os participantes. Além disso, trata-se de uma alternativa viável diante da recorrente ausência de usuários reais disponíveis para entrevistas nesse tipo de cenário, mas se faz necessário conhecimento implícito do time.

Conforme destacado por Barbosa et al. (2009), a escolha da técnica de elicitação deve considerar o contexto do projeto, e, em ambientes com forte limitação de tempo e ausência de

stakeholders formais, abordagens informais e criativas podem oferecer bons resultados. Ainda que simplificado, esse processo foi essencial para a definição dos requisitos do MarcaSUS, demonstrando que, mesmo fora de um ambiente ideal, é possível aplicar os fundamentos da engenharia de software de maneira adaptada e eficaz.

O MarcaSUS não está em produção e desde sua construção, em 2023, o protótipo funcional não foi mais utilizado. Como consequência da desmobilização da equipe após o evento, não foi possível recuperar o acesso ao código-fonte da aplicação móvel, impossibilitando a apresentação de suas telas e artefatos neste trabalho.

A participação da autora nesse processo representou um grande aprendizado em termos de desenvolvimento de sistemas e engenharia de *software*. A experiência adquirida ao longo do desenvolvimento do protótipo e deste trabalho foi fundamental para o aprimoramento das habilidades como analista de requisitos.

REFERÊNCIAS

Barbosa, G.; et al. Um processo de elicitação de requisitos com foco na seleção da técnica de elicitação. In: VIII Simpósio Brasileiro de Qualidade de Software (SBQS 2009). **Anais [...]**, Sociedade Brasileira de Computação (SBC), 2009.

Belgamo, A.; Martins, L. E. G Estudo Comparativo sobre as Técnicas de Elicitação de Requisitos do Software. In: XX Congresso Brasileiro da Sociedade Brasileira de Computação (SBC). **Anais [...]**, Curitiba, 2000.

Borges, G. O que significa hackathon?. **Agência USP de Inovação**, 2019. Disponível em: <https://www.inovacao.usp.br/o-que-significa-hackathon/>. Acesso em: 4 jul. 2025.

Caroli, P. **Lean Inception**: como alinhar pessoas e construir o produto certo. São Paulo: Caroli.org, 2018.

Changevision, Inc. **Astah**: Software Design Tool. [S. l.], 2025. Disponível em: https://astah.net/. Acesso em: 21 jul. 2025.

Cunha, A. A. Pacientes dormem em calçada de posto de saúde para conseguir fichas de exames e consultas em Natal. **G1**, Natal, 4 set. 2020. Disponível em: <https://g1.globo.com/rn/rio-grande-do-norte/noticia/2020/09/04/pacientes-dormem-em-calcada-de-posto-de-saude-para-conseguir-ficha-de-exames-e-consultas-em-natal.ghtml>. Acesso em: 01 de jul. de 2025.

Fagundes, R. M. **Engenharia de Requisitos**: Do perfil do Analista de Requisitos ao desenvolvimento de requisitos com UML e RUP. Salvador: [s.n.], 2011.

Fernandes, R. **Engenharia de software**. Curitiba: Fael, 2017.

Glinz, M.; et al. **Certified professional for requirements engineering**. [S.l.]: Betascript Publishing, 2010.

Guedes, G. T. A. **UML**: uma abordagem prática. São Paulo: Novatec, 2004.

Pfleeger, S. L. **Engenharia de software**: teoria e prática. 2. ed. São Paulo: Prentice Hall, 2004.

Instituto Brasileiro de Geografia e Estatística (IBGE). PNS 2019: sete em cada dez pessoas que procuram o mesmo serviço de saúde vão à rede pública. **Agência IBGE Notícias**, 2020. Disponível em: <https://agenciadenoticias.ibge.gov.br/agencia-sala-de-imprensa/2013-agencia-de-noticias/releases/28793-pns-2019-sete-em-cada-dez-pessoas-que-procuram-o-mesmo-servico-de-saude-vao-a-rede-publica#:~:text=Em%202019%2C%2076%2C5%25,que%20costumavam%20procurar%20mais%20frequentemente./>. Acesso em: 16 de JUN. de 2025.

Pressman, R. S. **Engenharia de Software**: uma abordagem profissional. 7. ed. São Paulo: McGraw-Hill, 2011.

Silva, R. C.; Benitti, F. B. V. **Introdução à Engenharia de Requisitos**: Relatório Técnico RT-ES-0001/11. São José, SC: Universidade do Vale do Itajaí, 2011.

Sommerville, I. **Engenharia de Software**. 9. ed. São Paulo: Pearson Prentice Hall, 2011.

Teixeira, K. Criminosos assaltam pessoas na fila da vacinação em Centro de Saúde na Grande Natal. **G1**, 2021. Disponível em: <https://g1.globo.com/rn/rio-grande-do-norte/noticia/2021/07/09/criminosos-assaltam-pessoas-na-fila-da-vacinacao-em-centro-de-saude-na-grande-natal.ghtml/>. Acesso em: 01 de jul. de 2025.