

MODELO REFERENCIAL DE REQUISITOS PARA APLICATIVOS DE DELIVERY: UM ESTUDO EXPLORATÓRIO BASEADO NA ISO/IEC 25010

REFERENCE REQUIREMENTS MODEL FOR DELIVERY APPLICATIONS: AN
EXPLORATORY STUDY BASED ON ISO/IEC 25010

Engenharias • 23/05/2026

REGISTRO DOI: [10.70773/revistatopicos/779464768](https://doi.org/10.70773/revistatopicos/779464768)

Rosaria Moraes Mariano¹

Auro Garcia²

Gioliano B Bertoni³

Wellington Ávila⁴

RESUMO

Atualmente, o crescente desafio das plataformas digitais de delivery obrigou mudanças significativas na estruturação dos requisitos de software, em especial nos ambientes móveis caracterizados por alta concorrência, exigências de desempenho e segurança e integração com múltiplos serviços. A norma ISO/IEC 25010 e suas diretrizes de qualidade fundamentam os princípios da Engenharia de Requisitos. O objetivo foi identificar, analisar e documentar requisitos funcionais e não funcionais essenciais, com base em revisão bibliográfica, análise de aplicativos líderes (iFood, Rappi) e validação com stakeholders reais. Assim, este estudo proporciona um modelo de referência de requisitos para aplicativos de delivery. Foram identificados e estruturados 19 requisitos funcionais e 9 requisitos não funcionais, segundo os critérios de qualidade e métricas mensuráveis. O modelo permite o desenvolvimento de produtos mínimos viáveis (MVP) para o setor, o que proporciona redução no retrabalho e ajuda a melhorar a qualidade de software em projetos futuros.

Palavras-chave: Engenharia de Requisitos; ISO/IEC 25010; Qualidade de Software; Requisitos Funcionais e Não Funcionais; Sistemas de Delivery.

ABSTRACT

Currently, the growing challenges of digital delivery platforms have required significant changes in the structuring of software requirements, especially in mobile environments characterized by high competition, performance and security demands, and integration with multiple services. The ISO/IEC 25010 standard and its quality guidelines provide the foundation for the principles of Requirements Engineering. The objective was to identify, analyze, and document essential functional and non-functional

requirements, based on a literature review, analysis of leading applications (iFood, Rappi), and validation with real stakeholders. Thus, this study provides a reference requirements model for delivery applications. A total of 19 functional requirements and 9 non-functional requirements were identified and structured according to quality criteria and measurable metrics. The model enables the development of Minimum Viable Products (MVPs) for the sector, which helps reduce rework and improve software quality in future projects.

Keywords: Requirements Engineering; ISO/IEC 25010; Software Quality; Functional and Non-Functional Requirements; Delivery Systems.

1. INTRODUÇÃO

O setor das plataformas digitais de entregas por aplicativos teve crescimento significativo que transformou profundamente as interações entre comércio e serviços, impulsionado pela necessidade e pela ampla adoção de dispositivos móveis. Os aplicativos de delivery tornaram-se o principal e estratégico meio de comunicação entre consumidores, estabelecimentos comerciais e entregadores.

A Engenharia de Requisitos consolidou-se como parte essencial da comunicação entre as diferentes stakeholders envolvidas no desenvolvimento de software, por conta do constante avanço tecnológico atual. Sendo assim, essa prática adiciona a produção de sistemas de qualidade, capazes de melhorar a produtividade organizacional e gerar ganhos aos negócios; contudo, quando o processo de levantamento e análise não é feito de modo apropriado,

podem demandar atrasos dispendiosos, retrabalho e expectativas não atendidas (Engholm Júnior, 2010).

2. FUNDAMENTAÇÃO TEÓRICA OU REVISÃO DA LITERATURA

Discutir requisitos em aplicativos de entrega exige olhar para dois pontos ao mesmo tempo: o que o sistema precisa fazer e como ele deve se comportar para entregar qualidade de fato. É justamente nesse cruzamento que a Engenharia de Requisitos se torna central, porque organiza as necessidades dos envolvidos e transforma essas necessidades em especificações claras para o desenvolvimento (Sommerville, 2011). Em projetos desse tipo, uma complexidade aparece desde o início. O aplicativo não atende apenas um usuário final: ele conecta clientes, restaurantes e entregadores, cada grupo com expectativas próprias. Além disso, o serviço depende de operações em tempo real, integração com pagamento digital, geolocalização, notificações e suporte. Sem requisitos bem definidos, o risco é construir funcionalidades desconectadas do uso real, com impacto direto na experiência e nos custos do projeto (Pressman; Maxim, 2016). A literatura de Engenharia de Requisitos mostra que bons resultados não vêm apenas de “levantar demandas”, mas de conduzir um processo estruturado. Kotonya e Sommerville (2000) apontam que a elicitação, análise, documentação e validação precisam ocorrer de forma articulada. Pohl e Rupp (2015) reforçam que os requisitos obrigatórios sejam compreensíveis, consistentes e testáveis, para reduzir ambiguidades entre as partes envolvidas. Wiegers e Beatty (2013), por sua vez, destacam a importância da rastreabilidade e da definição de critérios objetivos, permitindo verificar se o produto entregue corresponde ao problema que se pretende resolver. No caso específico de plataformas de entrega, os requisitos funcionais

compõem o fluxo básico de operação: cadastro, autenticação, busca de restaurantes, seleção de itens, fechamento de pedido, pagamento e envio da entrega. Esse conjunto sustenta o MVP e garante a operação mínima do serviço. Já ampliar requisitos funcionais complementares — como histórico de pedidos, cupons, filtros avançados, gestão para restaurantes e otimização de rotas para entregadores — tendem a eficiência operacional e valor percebido pelo usuário. Entretanto, o cumprimento das funcionalidades não é suficiente quando a proposta envolve uso frequente e alta concorrência. Os requisitos não funcionais são decisivos para o sucesso da solução, principalmente os relacionados a desempenho, disponibilidade, segurança, usabilidade, escalabilidade e manutenibilidade. A ISO/IEC 25010 (ISO, 2011) oferece uma base importante para organizar esses atributos e orientar sua medição. Em termos práticos, isso significa transformar qualidade em metas verificáveis, como tempo de resposta, nível de disponibilidade e padrões de proteção de dados. No contexto brasileiro, a dimensão legal também precisa ser incorporada desde a concepção. A Lei Geral de Proteção de Dados (Brasil, 2018) estabelece obrigações sobre coleta, armazenamento e tratamento de dados pessoais, o que impacta diretamente a definição de requisitos de segurança, privacidade e governança da informação. Com base nesse referencial, entende-se que um modelo de requisitos para aplicações de entrega deve integrar três eixos: necessidades dos stakeholders, critérios de qualidade de software e conformidade regulatória. Essa união fortalece o planejamento do MVP, reduz o retrabalho nas fases seguintes e cria condições mais sólidas para evolução do sistema em ambiente real de uso.

3. METODOLOGIA

Neste estudo, de abordagem exploratória, o objetivo foi compreender as necessidades pertinentes a aplicativos de delivery e lançar um modelo de requisitos para orientação em seu desenvolvimento.

A pesquisa foi orientada a partir de três estratégias combinadas: análise de aplicativos consolidados, com foco nas plataformas iFood e Rappi; validação prática, por meio de entrevistas semiestruturadas com cinco entregadores autônomos, aplicação de questionários a dez usuários frequentes (uso mínimo de quatro vezes por mês) e análise de 50 avaliações negativas no Google Play, buscando identificar falhas recorrentes e requisitos não atendidos. Baseou-se em bibliografia de autores consolidados em Engenharia de Software, como Pressman, Sommerville, Pohl e Wiegers.

3.1 Etapas do Levantamento de Requisitos

Seguindo o modelo adaptado de Sommerville (2011), o processo para o levantamento e análise de requisitos foi organizado nas etapas abaixo:

- i. identificação dos stakeholders: desenvolvedores e administradores, clientes, restaurantes parceiros e entregadores.
- ii. definição dos objetivos: rapidez nas entregas, aumentar vendas dos parceiros, otimizar rotas e manter as operações seguras.
- iii. extração de informações para os requisitos: entrevistas, questionários, análise de mercado e avaliações e *feedbacks* de usuários.

iv. análise e categorização: classificação dos requisitos em funcionais e não funcionais.

v. validação: verificação dos requisitos com base no retorno dos stakeholders envolvidos.

4. RESULTADOS E DISCUSSÃO

4.1. Requisitos Funcionais

Os requisitos funcionais definem as funcionalidades que o sistema deve executar para atender às necessidades dos stakeholders. Foram categorizados em essenciais (*core*), relacionados ao funcionamento básico do sistema, e complementares (diferenciais), agregando valor ao serviço.

Os requisitos funcionais essenciais garantem que o Produto Mínimo Viável (MVP) cumpra suas funções essenciais. Os complementares ampliam a experiência e a operação. A consolidação detalhada encontra-se nas Tabelas 1 e 2.

Tabela 1: Requisitos funcionais essenciais

ID	Descrição	Ator	Justificativa	Prioridade	Origem
RF01	Cadastro de Usuário (Cliente, Restaurant e, Entregador)	Todos	Acesso personalizado e seguro	Alta	Mercado
RF02	Login e	Todos	Controle	Alta	Mercado

⚠ Esta tabela possui muitas colunas e foi cortada para impressão. Para visualizá-la completa, acesse o artigo original em: <https://revistatopicos.com.br/artigos/modelo-referencial-de-requisitos-para-aplicativos-de-delivery-um-estudo-exploratorio-baseado-na-iso-iec-25010?noblockage>

Fonte: Adaptado de iFood/Rappi e validação com stakeholders. Mercado/Questionário

Tabela 2: Requisitos Funcionais Complementares

ID	Descrição	Ator	Justificativa	Prioridade	Origem
RF10	Filtros de busca (tipo, preço, nota, distância)	Cliente	Busca personalizada e otimizada	Alta	Mercado/Questionário
RF11	Histórico de	Cliente	Conveniência e	Alta	Mercado/Questionário

⚠ Esta tabela possui muitas colunas e foi cortada para impressão. Para visualizá-la completa, acesse o artigo original em: <https://revistatopicos.com.br/artigos/modelo-referencial-de-requisitos-para-aplicativos-de-delivery-um-estudo-exploratorio-baseado-na-iso-iec-25010?noblockage>

Fonte: Adaptado de iFood/Rappi e validação com stakeholders.

4.2. Requisitos Não Funcionais

Os requisitos não funcionais (RNF) definem as restrições do sistema e os atributos de qualidade que o sistema deve possuir. A categorização, para cobertura alinhada à norma, foi baseada na

ISO/IEC 25010 (ISO, 2011). O detalhamento mensurável encontra-se na Tabela 3.

Requisitos não funcionais

ID	Descrição	Categoria (ISO/IEC 25010)	Métrica Mensurável	Justificativa
RNF01	Disponibilidade do sistema 24 horas por dia, 7 dias por semana.	Confiabilidade	99,9% de Uptime mensal (máximo de 43 minutos de inatividade).	Garantir acesso contínuo ao serviço.
RNF02	Tempo de resposta para carregamento de telas e finalização de pedidos.	Desempenho	< 2 segundos em 95% das requisições (latência máxima).	Experiência fluida e satisfatória do usuário
RNF03	Criptografia de dados sensíveis (senhas, informações de pagamento).	Segurança	Uso de TLS 1.3 e criptografia AES-256 para dados em repouso	Proteção contra acesso não autorizado
RNF04	Conformidade com a Lei Geral de Proteção de Dados (LGPD).	Legalidade	Auditoria de compliance anual e política de privacidade clara	Atendimento à legislação brasileira (Lei nº 13.709/2018).

RNF05	Compatibilidade com os sistemas operacionais móveis líderes	Portabilidade	Suporte às duas últimas versões do Android e iOS	Amplo alcance de usuários
RNF06	Interface intuitiva e de fácil aprendizado	Usabilidade	Taxa de sucesso de 90% em testes de usabilidade (primeiro pedido).	Facilidade de uso para todos os stakeholders
RNF07	Suporte a um grande volume de usuários simultâneos	Escalabilidade	Suporte a 10.000 usuários simultâneos com tempo de resposta RNF02.	Crescimento sustentável do negócio
RNF08	Manutenibilidade do código e da arquitetura	Manutenibilidade	Cobertura de testes unitários > 80% e documentação da arquitetura	Facilidade de correção de bugs e evolução do sistema
RNF09	Requisitos de Backup e recuperação de dados	Confiabilidade	Backup diário com RTO (Recovery Time Objective) de 4 horas.	Prevenção de perda de dados.

Fonte: Elaborado pelos autores, com base em Sommerville e ISO/IEC 25010.

4.3. Restrições do Projeto

Além dos requisitos funcionais e não funcionais, o projeto também precisa considerar algumas restrições e diretrizes extras que influenciam diretamente seu desenvolvimento:

- prazo: o MVP precisa estar pronto em até seis meses, o que exige planejamento ágil e bem estruturado.
- custo: o orçamento deverá ser proposto conforme as necessidades do projeto e das demandas dos stakeholders.
- interface: o design deve seguir as diretrizes do Material Design (Android) e do Human Interface Guidelines (iOS), garantindo experiência familiar e intuitiva.
- tecnologia: o *backend* será construído em Python, com Django ou FastAPI, e o *frontend* com React Native, permitindo desempenho consistente em Android e iOS.

4.4. Resposta à Questão de Pesquisa

Sim, é possível desenvolver um modelo referencial de requisitos aplicável a novos aplicativos de delivery voltados ao mercado brasileiro. A proposta reúne 19 requisitos funcionais (RF) e 9 requisitos não funcionais (RNF), conforme o padrão ISO/IEC 25010. Ele funciona como um *checklist* consistente e prático para apoiar a criação de um MVP.

4.5. Contribuições do Estudo

Do ponto de vista prático, o modelo contribui para reduzir incertezas nas fases iniciais do projeto, auxiliando equipes no direcionamento do desenvolvimento e diminuindo retrabalho. No campo teórico, o estudo reforça a aplicação de conceitos da Engenharia de Requisitos, articulando critérios de qualidade propostos por Wiegers com padrões internacionais de qualidade de software.

4.6. Limitações

A principal limitação está no tamanho da amostra, composta por cinco entregadores e dez usuários, selecionados por conveniência. Além disso, o modelo ainda não foi validado em um ciclo completo de desenvolvimento, incluindo implementação e testes.

4.7. Trabalhos Futuros

As pesquisas futuras devem concentrar-se na validação empírica do modelo proposto, por meio do desenvolvimento de um MVP baseado nos requisitos descritos e da análise de métricas de desempenho satisfação do usuário, disponibilidade (*uptime*) e tempo de resposta em ambiente de produção real.

5. CONCLUSÃO/CONSIDERAÇÕES FINAIS

A Engenharia de Requisitos é o alicerce para o sucesso de aplicativos de delivery competitivos e dinâmicos. Este estudo identificou, analisou e documentou um modelo referencial de requisitos, proposto com base em estudos anteriores e nas opiniões de pessoas que vivenciam a experiência de uso e operação.

REFERÊNCIAS BIBLIOGRÁFICAS

BRASIL. Lei no 13.709, de 14 de agosto de 2018. Lei Geral de Proteção de Dados Pessoais (LGPD). Diário Oficial da União, Brasília, DF, 15 ago. 2018.

ENGHOLM JÚNIOR, A. A. Engenharia de Requisitos: uma abordagem prática. Rio de Janeiro: LTC, 2010.

ISO/IEC25010:2011–Systems and software engineering Systems and software Quality Requirements and Evaluation (SQuaRE) System and software quality models. Geneva: ISO, 2011.

KOTONYA, Gerald; SOMMERVILLE, Ian. Requirements engineering: processes and techniques. New York: Wiley, 2000.

POHL, Klaus; RUPP, Chris. Requirements Engineering Fundamentals. 3. ed. Rocky Nook, 2015.

PRESSMAN, Roger S.; MAXIM, Bruce R. Engenharia de software: uma abordagem profissional. 8. ed. Porto Alegre: AMGH, 2016.

SOMMERVILLE, Ian. Engenharia de software. 9. ed. São Paulo: Pearson, 2011.

WIEGERS, Karl E.; BEATTY, Joy. Software Requirements. 3. ed. Microsoft Press, 2013.

¹ Psicóloga clínica, Técnica de montagem e manutenção de computadores, discente do curso de Engenharia de Software da Universidade de Vassouras Campus Saquarema. ORCID: <https://orcid.org/0009-0009-4083-306X>. E-mail: [acesse o artigo original para visualizar o e-mail](#)

² Técnico de Contabilidade, técnico em Agrimensura, discente do curso de Engenharia de Software da Universidade de Vassouras Campus Saquarema. ORCID: <https://orcid.org/0009-0007-4069-021X>.

E-mail: [acesse o artigo original para visualizar o e-mail](#)

³ Formado em Tecnologia em Sistemas de Computação UFF. Com pós em Criptografia UFF/IME Professor Universidade de Vassouras.

ORCID: <https://orcid.org/0009-0003-4931-3745>. E-mail: [acesse o artigo original para visualizar o e-mail](#)

⁴ Mestre em Gestão do Trabalho para a Qualidade do Ambiente Construído Universidade Santa Úrsula (USU) Professor Universidade de Vassouras. ORCID: <https://orcid.org/0000-0002-9377-5993>. E-mail:

[acesse o artigo original para visualizar o e-mail](#)