



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA

Augusto Parigot de Souza

**Proposta de um modelo e processo para desenvolvimento de produto
multidisciplinar**

Florianópolis
2024

Augusto Parigot de Souza

**Proposta de um modelo e processo para desenvolvimento de produto
multidisciplinar**

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Mecânica da Universidade Federal de Santa Catarina como requisito parcial para a obtenção do título de Mestre em Projeto de Sistemas Mecânicos

Orientador: Prof. Dr. Eng. Rodrigo Bastos Fernandes

Florianópolis

2024

Parigot de Souza, Augusto

Proposta de um modelo e processo para desenvolvimento de produto multidisciplinar / Augusto Parigot de Souza ; orientador, Rodrigo Bastos Fernandes, 2024.
169 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia Mecânica, Florianópolis, 2024.

Inclui referências.

1. Engenharia Mecânica. 2. Desenvolvimento de produtos. 3. Produto multidisciplinar. 4. Engenharia de Sistemas. I. Bastos Fernandes, Rodrigo . II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia Mecânica. III. Título.

Augusto Parigot

**Proposta de um modelo e processo para desenvolvimento de produto
multidisciplinar**

O presente trabalho em nível de Mestrado foi avaliado e aprovado, em 01 de novembro de 2024, pela banca examinadora composta pelos seguintes membros:

Prof. Dr. Eng. André Ogliari
Universidade Federal de Santa Catarina

Prof. Dr. Eng. Julio Cesar Frantz
UNIVALI

PhD. Eng. Luís Gonzaga Trabasso
Instituto SENAI de Inovação

Certificamos que esta é a versão original e final do trabalho de conclusão que foi julgado adequado para obtenção do título de Mestre em Engenharia Mecânica

Insira neste espaço a
assinatura digital

Coordenação do Programa de Pós-Graduação

Insira neste espaço a
assinatura digital

Prof. Dr. Eng. Rodrigo Bastos Fernandes
Orientador

Florianópolis, 2024.

AGRADECIMENTOS

Agradeço a todos que me acompanharam durante essa jornada. Tal jornada foi iniciada e não seria possível sem os colegas e amigos nedipianos Guilherme Loureiro, Rafael Veras, Natalia Boelter, Heitor Kagueiama, Fabíola Reinert, Rogério Lampert, Willian Moldenhauer, Paulo Henrique Zen, Edson Filho e Lucas Yoshida.

Aos companheiros de ISI, em especial Arthur Weschenfelder, Ísis Teixeira, Renato Simão, Fabiano Preisler, Willian Henrique e os parceiros de luta na concepção e montagem de bancadas, satélites, sensores e robôs (títulos omitidos).

Aos amigos de outras jornadas: Isadora Fontenelle, Carlos Mezzaroba, Lucas Pimentel, Felipe Pires, Michele Honicky, Marília Duarte, João Miranda, Diego Capeletti e Vinicius Poloni.

Aos coordenadores, servidores e professores do POSMEC, em especial ao meu orientador Rodrigo Bastos Fernandes pelos ensinamentos e direcionamento rumo ao final do percurso.

À Fernanda, pelo carinho, apoio e parceria na jornada. E à minha família, por sempre me proporcionar o necessário nos rumos escolhidos.

“If I had only one hour to solve a problem, I would spend up to two-thirds of that hour in attempting to define what the problem is.” Atribuído a William H. Markle. “*The Manufacturing Manager’s Skills*” (1966)

RESUMO

Atualmente, uma grande gama de produtos possui eletrônica embarcada, nos quais software e componentes físicos estão profundamente integrados e são interdependentes. Esses produtos podem ser considerados multidisciplinares, pois envolvem conhecimentos de distintas áreas, ou disciplinas de engenharia, tais como engenharia mecânica, engenharia eletrônica, engenharia de software, entre outras. O referencial teórico levantado, juntamente com a realização de uma pesquisa exploratória, identificou que os processos tradicionais de desenvolvimento de produtos podem não ser adequados para produtos multidisciplinares, pois cada uma das áreas de conhecimento envolvidas apresenta suas particularidades, ciclos e processos de desenvolvimento distintos. Ademais, resultante dessa pesquisa, identificou-se a engenharia de sistemas como apropriada para integração de áreas de conhecimento, ainda que não guie o desenvolvimento das áreas propriamente dito. Assim, este trabalho propõe um modelo e processo para o desenvolvimento de um produto multidisciplinar com a integração de produtos de três disciplinas de engenharia, tais quais desenvolvimento mecânico, desenvolvimento eletrônico e desenvolvimento de software. O Modelo de Desenvolvimento de Produto Multidisciplinar (MDPM) é construído utilizando processos advindos do Processo de Desenvolvimento de Produtos (PDP), boas práticas de desenvolvimento eletrônico e modelos ágeis de desenvolvimento de software, integrados conforme os preceitos da engenharia de sistemas. A avaliação do MDPM se deu conforme três contextos: acadêmico - aplicação em projeto de P&D; contexto empresarial – desenvolvedores de produtos criativos; e por especialistas em desenvolvimento de produtos. Conforme as avaliações, o modelo proposto se apresenta adequado, sendo capaz de guiar equipes no processo de desenvolvimento de produtos multidisciplinares, favorecendo a integração e comunicação entre as disciplinas de engenharia envolvidas, estabelecendo documentação e evitando retrabalho.

Palavras-chave: Desenvolvimento de produtos; Produto multidisciplinar; Desenvolvimento mecânico; Desenvolvimento eletrônico; Desenvolvimento de software; Engenharia de sistemas.

ABSTRACT

Nowadays, many products are built with embedded electronics, in which software and physical components are deeply integrated and interdependent. Those products can be called multidisciplinary, as they encompass information from different fields of knowledge, or disciplinary engineering, such as mechanical engineering, electronic engineering, and software engineering. The literature review and exploratory research have identified that the traditional product development processes may not be suitable for multidisciplinary products, as each field of knowledge involved has its particularities, cycles, and distinct development processes. Furthermore, as a result of this research, systems engineering was identified as appropriate for integrating the fields of knowledge, even though it does not guide the development in the fields of knowledge involved. Hence, this master's thesis proposes a reference model for multidisciplinary product development with the integration of products from three disciplinary engineering, such as mechanical, electronic, and software development. The Multidisciplinary Product Development Model (MDPM) is made up of processes from the Product Development Process (PDP), good practices from electronic development, and agile models from software development, all of which are integrated according to the precepts from systems engineering. The evaluation of the MDPM was performed in three contexts: academical – practical application on an R&D project; business context – creative product developers; and product development specialists. According to the evaluation, the proposed model seems adequate, being suitable for guiding teams in multidisciplinary product development as it enhances integration and communication between the disciplinary engineering, establishes documentation, and helps avoid rework.

Keywords: Product design; Multidisciplinary product; Mechanical development; Electronic development, Software development; Systems Engineering

LISTA DE FIGURAS

Figura 1 - Exemplos de produtos.	26
Figura 2 - Fases do PDP segundo Asimov.	29
Figura 3 - Fases do PDP segundo Pahl e Beitz.	30
Figura 4 - Modelo Total Design.	31
Figura 5 - Modelo de Ulrich e Eppinger.	32
Figura 6 - Fases de Rozenfeld <i>et al.</i>	32
Figura 7 - Modelo PRODIP.	33
Figura 8 - Estrutura funcional de acordo com grau de inovação.	37
Figura 9 - Matriz Morfológica.	38
Figura 10 - Configuração típica de uma PCI.	44
Figura 11 - Esquemático de uma PCI.	46
Figura 12 - Layout de uma PCI.	47
Figura 13 - Passos para o projeto de PCI.	48
Figura 14 - Processo do Ciclo de Vida do Software.	52
Figura 15 - Exemplo de diagrama UML.	54
Figura 16 - Modelo Cascata.	58
Figura 17 - Modelo da Prototipação.	59
Figura 18 - Modelo Espiral.	59
Figura 19 - Modelo RUP.	60
Figura 20 - Modelo V.	61
Figura 21 - Modelo Scrum.	63
Figura 22 - Modelo XP.	64
Figura 23 - Quadro Kanban.	65
Figura 24 - Ciclo DEVOPS.	66
Figura 25 - Diagramas do SysML.	72
Figura 26 - Processo de ES da NASA.	77
Figura 27 - Locomotiva da ES.	78
Figura 28 - Fases e tarefas-chave no processo do INCOSE.	79
Figura 29 - Modelo V da ES.	80
Figura 30 - Representação do MDPM.	91
Figura 31 - Diagrama de Origem do MDPM.	96
Figura 32 - Relação entre DEs e ES.	97

Figura 33 - Atividades da Engenharia de Sistemas.....	99
Figura 34 - Atividades de Desenvolvimento Mecânico.....	101
Figura 35 - Atividades de Desenvolvimento Eletrônico.....	103
Figura 36 - Atividades de Desenvolvimento de Software.....	105
Figura 37 - Respostas aos critérios em contexto de projeto.....	111
Figura 38 - Avaliação do MPDM em contexto de projeto.....	112
Figura 39 - Respostas aos critérios em contexto empresarial.....	116
Figura 40 - Avaliação do MDPM em contexto empresarial.....	117
Figura 41 - Respostas dos especialistas aos critérios estabelecidos.....	122
Figura 42 - Avaliação do MDPM por especialistas.....	123
Figura 43 - Conteúdo das entrevistas.....	137
Figura 44 - Processo de desenvolvimento dos pesquisadores de desenvolvimento mecânico.....	137
Figura 45 - Processo de desenvolvimento dos pesquisadores de desenvolvimento eletrônico.....	138
Figura 46 - Processo de desenvolvimento dos pesquisadores de desenvolvimento de software.....	138
Figura 47 - Fluxos do processo.....	159

LISTA DE QUADROS

Quadro 1 - Domínios do MDPM.	88
Quadro 2 - Relação das fases entre os modelos.	89
Quadro 3 - Revisões de Produto do MDPM.	92
Quadro 4 - Revisões integradas do MDPM.	93
Quadro 5 - Questionário de avaliação por equipe de projeto	110
Quadro 6 - Questões abertas da equipe de projeto.	112
Quadro 7 - Questionário para contexto empresarial.	115
Quadro 8 - Comentários e sugestões da equipe técnica.	117
Quadro 9 - Perfil dos Avaliadores Especialistas.	119
Quadro 10 - Questionário do critério AUC.	120
Quadro 11 - Questionário do critério Relevância.	120
Quadro 12 - Questionário do critério Eficiência Econômica.	121
Quadro 13 - Questionário do critério Clareza.	121
Quadro 14 - Questionário do critério Estruturação Sistemática.	121
Quadro 15 - Questionário do critério Adaptabilidade.	121
Quadro 16 - Questionário do critério Generalidade.	122
Quadro 17 - Comentários e sugestões dos especialistas ao modelo de referência.	124
Quadro 18 - Entregáveis da equipe de Mecânica.	140
Quadro 19 - Entregáveis da equipe de Eletrônica.	140
Quadro 20 - Entregáveis da equipe de Software.	141
Quadro 21 - Valores da abordagem ágil.	164
Quadro 22 - Princípios da abordagem ágil.	165
Quadro 23 - Revisões da Engenharia de Sistemas.	166

LISTA DE TABELAS

Tabela 1 -Perfil da equipe técnica avaliadora em contexto de Projeto P&D.	110
Tabela 2 - Perfil da equipe técnica avaliadora em contexto empresarial.	114
Tabela 3 - Conhecimento prévio da equipe avaliadora em modelos de desenvolvimento	115
Tabela 4 - Modelos mencionados.	139
Tabela 5 - Sugestões para Modelos Multidisciplinares.	142

LISTA DE ABREVIATURAS E SIGLAS

ANEEL	Agência Nacional de Energia Elétrica
ANSI	<i>American National Standards Institute</i>
ASME	<i>American Society of Mechanical Engineering</i>
AUC	<i>Accuracy, Unambiguity and Completeness</i>
BoM	<i>Bill of Materials</i>
CAD	<i>Computer Aided Design</i>
CAE	<i>Computer Aided Engineering</i>
CDR	<i>Critical Design Review</i>
CERR	<i>Critical Event Readiness Review</i>
ConOps	<i>Concept of Operations</i>
COTS	<i>Commercial-off-the-shelf</i>
DE	Disciplinas de Engenharia
DevOps	<i>Development & Operations</i>
DR	<i>Decommissioning Review</i>
DRR	<i>Disposal Readiness Review</i>
EDA	<i>Electronic Design Automation</i>
EIA	<i>Electronic Industries Alliance</i>
EPROM	<i>Erasable Programmable ROM</i>
ES	Engenharia de Sistemas
FPGA	Field Programmable Gate Array
FR-4	<i>Flame Retardant class 4</i>
IDEF0	<i>Integration Definition for Process Modelling</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
INCOSE	<i>International Council on Systems Engineering</i>
ISO	<i>International Organization for Standardization</i>
MCR	<i>Mission Concept Review</i>
MDR	<i>Mission Definition Review</i>
MDPM	Modelo de Desenvolvimento de Produto Multidisciplinar
NASA	<i>National Aeronautics and Space Administration</i>
ORR	<i>Operational Readiness Review</i>
P&D	Pesquisa e Desenvolvimento
PBS	<i>Product Breakdown Structure</i>

PCI	Placa de Circuito Impresso
PD&I	Pesquisa, Desenvolvimento e Inovação
PDE	Processo de Desenvolvimento Eletrônico
PDP	Processo de Desenvolvimento de Produtos
PDR	<i>Preliminary Design Review</i>
PLAR	<i>Post-Launch Assessment Review</i>
PRODIP	Processo de Desenvolvimento Integrado de Produtos
QFD	<i>Quality Function Deployment</i>
RAP	Revisão de Arquitetura do Produto
RAPM	Revisão de Aceitação do Produto Multidisciplinar
RCP	Revisão de Conceito do Produto
RIA	Revisão Integrada de Arquitetura
RICC	Revisão Integrada Crítica de Conceitos
RIO	Revisão Integrada de Objetivos
RIPC	Revisão Integrada Preliminar de Conceitos
RIPM	Revisão Integrada de Produto Multidisciplinar
RIR	Revisão Integrada de Requisitos
RIRP	Revisão Integrada de Realização de Produtos
ROM	<i>Read-Only Memory</i>
RRP	Revisão de Requisitos do Produto
RTP	Revisão de Testes do Produto
RUP	<i>Rational Unified Process</i>
RVPM	Revisão de Verificação do Produto Multidisciplinar
SAR	<i>System Acceptance Review</i>
SDR	<i>System Definition Review</i>
SIR	<i>System Integration Review</i>
SLCP	<i>Software Life Cycle Process</i>
SRR	<i>System Requirements Review</i>
SysML	<i>Systems Modeling Language</i>
TRL	<i>Technology Readiness Level</i>
UI	<i>User Interface</i>
UML	<i>Unified Modeling Language</i>
VDI	<i>Verein Deutscher Ingenieure</i>
XP	<i>Extreme Programming</i>

SUMÁRIO

1	INTRODUÇÃO	17
1.1	CONTEXTUALIZAÇÃO.....	17
1.2	PROBLEMAS DE PESQUISA.....	18
1.3	OBJETIVOS.....	20
1.4	JUSTIFICATIVA.....	21
1.5	METODOLOGIA DE PESQUISA.....	21
1.6	ESTRUTURA DO DOCUMENTO.....	22
2	REFERENCIAL TEÓRICO	24
2.1	PROCESSO DE DESENVOLVIMENTO DE PRODUTOS.....	24
2.1.1	Histórico	25
2.1.2	Definições	25
2.1.3	Modelos de PDP	28
2.1.3.1	<i>Projeto Informacional</i>	34
2.1.3.2	<i>Projeto Conceitual</i>	36
2.1.3.3	<i>Projeto Preliminar</i>	39
2.1.3.4	<i>Projeto Detalhado</i>	41
2.1.4	Considerações sobre o Processo de Desenvolvimento de Produtos ..	42
2.2	DESENVOLVIMENTO ELETRÔNICO.....	43
2.2.1	Definições	43
2.2.2	Processo de desenvolvimento	44
2.2.3	Considerações sobre o Desenvolvimento Eletrônico	48
2.3	DESENVOLVIMENTO DE SOFTWARE.....	49
2.3.1	Definições	50
2.3.2	Processo de desenvolvimento	51
2.3.2.1	<i>Levantamento de Requisitos</i>	53
2.3.2.2	<i>Modelagem</i>	54
2.3.2.3	<i>Construção</i>	56
2.3.2.4	<i>Entrega</i>	57
2.3.3	Modelos tradicionais	57
2.3.4	Modelos ágeis	62
2.3.5	Considerações sobre o Desenvolvimento de Software	66
2.4	ENGENHARIA DE SISTEMAS.....	67

2.4.1	Processos técnicos	68
2.4.1.1	<i>Definição de Expectativas e Requisitos de Stakeholders.....</i>	69
2.4.1.2	<i>Definição de Requisitos do Sistema.....</i>	70
2.4.1.3	<i>Definição de Arquitetura.....</i>	71
2.4.1.4	<i>Definição de Projetação</i>	72
2.4.1.5	<i>Implementação.....</i>	73
2.4.1.6	<i>Integração</i>	74
2.4.1.7	<i>Verificação.....</i>	75
2.4.1.8	<i>Validação.....</i>	76
2.4.2	Modelo da NASA	76
2.4.3	Modelo do INCOSE	78
2.4.4	Modelo V de Desenvolvimento	80
2.4.5	Considerações da Engenharia de Sistemas	81
2.5	CONSIDERAÇÕES DO CAPÍTULO	81
3	PESQUISA DE CAMPO E PROPOSIÇÃO DA METODOLOGIA.....	84
3.1	RESTRIÇÃO DE ESCOPO	84
3.2	PROPOSTA DO MODELO DE DESENVOLVIMENTO DE PRODUTO MULTIDISCIPLINAR.....	86
3.2.1	Ciclo de desenvolvimento do MDPM.....	87
3.2.2	Apresentação do MDPM	90
3.2.2.1	<i>Revisões do produto</i>	92
3.2.2.2	<i>Revisões integradas.....</i>	93
3.2.2.3	<i>Processos do MDPM.....</i>	95
3.3	CONSIDERAÇÕES DO CAPÍTULO	106
4	APLICAÇÃO E AVALIAÇÃO DO MDPM.....	107
4.1	APLICAÇÃO EM PROJETO DE P&D	107
4.1.1	Estágio do projeto no momento da aplicação.....	108
4.2	AVALIAÇÃO.....	109
4.2.1	Contexto de projeto P&D.....	109
4.2.2	Contexto empresarial	114
4.2.3	Avaliação por parte de Especialistas	118
4.3	CONSIDERAÇÕES DO CAPÍTULO	126
5	CONCLUSÃO	127
5.1	CONSIDERAÇÕES E LIMITAÇÕES.....	127

5.2	SUGESTÕES PARA TRABALHOS FUTUROS	130
	REFERÊNCIAS	131
	APÊNDICE A – PESQUISA EXPLORATÓRIA	136
	APÊNDICE B – QUESTIONÁRIO: PDP APLICADO A PROJETOS MULTIDISCIPLINARES	143
	APÊNDICE C – REVISÕES DO PRODUTO	145
	APÊNDICE D – REVISÕES INTEGRADAS	149
	APÊNDICE E – FLUXOS DO PROCESSO	158
	ANEXO A – BOAS PRÁTICAS NO PROJETO DE PCIS	162
	ANEXO B – VALORES DA ABORDAGEM ÁGIL	164
	ANEXO C – PRINCÍPIOS DA ABORDAGEM ÁGIL.....	165
	ANEXO D – REVISÕES DA ENGENHARIA DE SISTEMAS	166

1 INTRODUÇÃO

Este capítulo contextualiza as motivações e problemas de pesquisa, com intuito de explorar o desenvolvimento de produto multidisciplinar. Para tanto, além da contextualização, através de um breve histórico, e a formulação da problemática de pesquisa, são apresentados os objetivos, justificativa e procedimentos metodológicos.

1.1 CONTEXTUALIZAÇÃO

Na era pré-Revolução Industrial o modelo de produção da sociedade se constituiu de especialistas como alfaiates, ferreiros e outros artesãos, os quais fabricavam produtos e artefatos de acordo com as necessidades. Não havia uma distinção clara entre projeto e produção.

Na Primeira e Segunda Revoluções Industriais (séculos XVIII e XIX), e com o surgimento das fábricas, o panorama mudou: os produtos tornaram-se em sua maioria produzidos em massa, reduzindo os custos e levando à adoção da padronização de componentes – em oposição ao sistema artesanal, no qual cada produto é único (Back *et al.*, 2008).

Posteriormente, no século XX, houve a Terceira Revolução Industrial, caracterizada por uma maior aplicação da ciência incorporada à produção nos campos da robótica, informática, telecomunicações, eletrônica, entre outros (Morgan e Liker, 2008).

Cada uma das revoluções industriais, juntamente com a globalização da economia, trouxe consigo um novo panorama, com aumento de competitividade, redução de custos e abertura de novos mercados. Novos mercados exigem maior qualidade de produtos e são impulsionados por novidades, por inovação. Para suprir essas demandas, se fez necessário sistematizar e otimizar a maneira como soluções são desenvolvidas e disponibilizadas, consolidando-se, assim, o processo de desenvolvimento de produtos (PDP). O PDP tem como função estruturar o planejamento, execução e controle do desenvolvimento de produtos por meio de uma série de atividades sistematizadas, suportadas por métodos, técnicas e ferramentas (Loureiro, 2013).

Segundo Back *et al.* (2008), a aplicação de procedimentos bem estabelecidos traz diversos benefícios, como a redução de tempo de

desenvolvimento, de modificações e promove o aumento da qualidade em diversos aspectos. Apesar desses benefícios, muitas das ferramentas, métodos ou processos encontrados na literatura de desenvolvimento de produtos são dirigidos a profissionais de engenharia mecânica (Pagan; Silva; Mello, 2013).

Em contrapartida, com o advento da revolução digital, parte da contemporânea Quarta Revolução Industrial, um número maior de produtos apresenta e requer eletrônica embarcada, nos quais software e componentes físicos são interdependentes. Esses produtos podem ser considerados multidisciplinares, pois abrangem desenvolvimento de diversos campos do conhecimento (Khaitan e McCalley, 2015).

Para Guérineau *et al.* (2022), a crescente multidisciplinaridade obriga empresas a adaptar suas práticas de desenvolvimento, pois os caminhos tradicionais para desenvolver produtos podem não ser adequados para lidar com a grande variedade de disciplinas envolvidas.

O desenvolvimento de produtos, seja para a indústria ou para o consumidor direto, pode envolver conhecimentos de distintas áreas: engenharia mecânica, engenharia eletrônica, controle e automação, ciências da computação, engenharia de software, entre outras. Dessa maneira, os processos de desenvolvimento carecem de novas práticas, sobretudo quando agregam áreas do conhecimento cujo processo de desenvolvimento exige iterações, testes ou incrementos constantes, como o caso de placas eletrônicas e software (Pagan; Silva; Mello, 2013).

1.2 PROBLEMAS DE PESQUISA

Identificam-se na literatura modelos para o PDP que embasam e apoiam o desenvolvimento de produtos físicos por meio de uma sequência de fases, atividades e entregas, caracterizados por extensa exploração, preparação e decisões atribuídas no início do desenvolvimento. Contudo, para produtos intangíveis como, por exemplo, a criação de um software, o processo de desenvolvimento envolve ciclos de implementação, em que são entregues incrementos (desenvolvimento incremental), visando rápida adaptação e entrada no mercado. Tais processos são significativamente diferentes entre si, utilizam vocabulários distintos e diferentes técnicas de gerenciamento (Hendler, 2019).

Uma vez que muitos produtos possuem ambas as características (tangível e intangível), pode-se indagar qual é o processo mais indicado nesses casos. Hendler (2019) aponta que coordenar práticas de desenvolvimentos distintas ainda não é um processo bem entendido e explorado na literatura. Segundo Mule *et al.* (2020), a indústria demanda uma abordagem mais flexível e ágil para desenvolvimento de produtos multidisciplinares, pois os processos tradicionais sequenciais são ineficientes para as demandas atuais, especialmente as aplicações em sistemas ciberfísicos ou indústria 4.0. A utilização de modelos ágeis também é apontada por Michalides *et al.*, (2023) no desenvolvimento de produtos industriais, que ressaltam que uma adoção direta de práticas ágeis de software em produtos físicos resulta em novos desafios. Um desses desafios, segundo Hendler (2021) é o de coordenar processos que são adaptáveis e estáveis, em produtos compreendendo partes tangíveis e intangíveis, especialmente em produtos que há necessidade de troca constante de informações entre as partes.

Além da tangibilidade, a indagação pode se estender para distintas áreas do conhecimento, formando um produto não somente tangível e intangível, mas multidisciplinar. O envolvimento de diversas áreas de conhecimento no processo de desenvolvimento de soluções requer ainda uma adequação no que diz respeito às técnicas e aos vocabulários, pois, quando o desenvolvimento envolve muitos parâmetros, a comunicação e colaboração entre diferentes áreas se torna muito demorada e conflituosa. A engenharia de sistemas foi proposta para prover uma plataforma de comunicação que garanta que cada área cumpra suas necessidades. Porém, Segundo Mcharek *et al.* (2019) e Guérineau (2022), há uma fragmentação existente entre a engenharia de sistemas as disciplinas de engenharia, o que traz a necessidade do estabelecimento de um melhor “caminho” de interação.

Diante do exposto, as seguintes perguntas de pesquisa são estabelecidas:

- i. As abordagens existentes no PDP são capazes de suportar práticas e processos de desenvolvimento de produtos de diferentes áreas do conhecimento?
- ii. É possível definir e especificar ferramentas, diretrizes e entregas específicas, por área de conhecimento envolvida, mantendo-se o

alinhamento entre essas, para o desenvolvimento de produto multidisciplinar?

- iii. Como estabelecer a integração e comunicação necessária entre os desenvolvimentos de diferentes áreas de conhecimento para assegurar o desenvolvimento integrado de um produto multidisciplinar?

A partir das questões de pesquisa traçadas podem-se estabelecer os objetivos geral e específicos da presente pesquisa, como apontado na seção seguinte.

1.3 OBJETIVOS

O objetivo geral deste trabalho consiste em propor um modelo e processo para o desenvolvimento de produtos multidisciplinares capaz de integrar e compatibilizar as particularidades dos processos de desenvolvimento de cada área do conhecimento envolvida.

Tal objetivo pode ser alcançado com auxílio dos seguintes objetivos específicos:

- Organizar o conhecimento encontrado na literatura sobre o processo de desenvolvimento de produtos de diferentes áreas de conhecimento;
- Explorar a literatura acerca da engenharia de sistemas como meio de integração de produtos de diferentes áreas do conhecimento;
- Realizar pesquisa exploratória com projetistas de diferentes áreas do conhecimento acerca de seus processos de desenvolvimento, e experiência no desenvolvimento de produto multidisciplinar;
- Propor um modelo e processo que considerem o desenvolvimento de produtos multidisciplinares;
- Sugerir atividades, fases e critérios de passagem entre essas, para o desenvolvimento multidisciplinar de produtos;
- Selecionar ferramentas que auxiliem na execução de atividades necessárias para o desenvolvimento de soluções de acordo com as áreas do conhecimento;
- Aplicar a proposição do modelo em um caso;
- Avaliar o modelo proposto.

1.4 JUSTIFICATIVA

O conhecimento estruturado, ou padronizado, é de grande valia às empresas por ser de fácil acesso e utilização na tomada de decisões (Mcharek *et al.*, 2019). Isso é especialmente verdadeiro em processos pouco previsíveis, como o desenvolvimento de produtos. Segundo Morgan e Liker (2008), “...um processo padronizado de desenvolvimento significa padronizar tarefas comuns, sequências e duração de tarefas e utilizar tudo isso como a base para a melhoria contínua do processo de desenvolvimento de produtos.”

A proposição do modelo de desenvolvimento busca auxiliar projetistas e escritórios de projeto em:

- Identificar relações e integrações nas áreas de conhecimento envolvidas no desenvolvimento de produtos multidisciplinares;
- Aumentar o entendimento dos projetistas acerca do desenvolvimento de outras áreas conjuntas;
- Auxiliar os projetistas a entenderem a importância e influência do desenvolvimento de sua área em um produto multidisciplinar;
- Estabelecer processos mais previsíveis de desenvolvimento de produtos multidisciplinares;
- Identificar e reduzir os retrabalhos no desenvolvimento de produtos multidisciplinares;
- Padronizar critérios de decisão para avanços de fase;
- Melhorar a qualidade dos produtos desenvolvidos.

1.5 METODOLOGIA DE PESQUISA

O presente trabalho se define tanto como uma pesquisa exploratória, quanto como uma aplicação em campo caracterizada por investigações realizadas por meio da coleta de dados junto a projetistas, profissionais e especialistas.

Em um primeiro momento, a fim de esclarecer, organizar e explorar a literatura nos assuntos estabelecidos nos objetivos específicos, são utilizados dois recursos principais: livros técnicos para embasamento do conteúdo de cada área do conhecimento distinta e artigos científicos para assegurar a relevância e atualidade do conhecimento.

Para a pesquisa exploratória, é realizada a análise do contexto de uma empresa de desenvolvimento de produtos multidisciplinares inovadores sob encomenda. Tal pesquisa é feita de forma anônima, com o objetivo do aprofundamento do conhecimento levantado com a prática profissional, sobretudo a fim de explorar a experiência prévia dos projetistas quanto a modelos, ferramentas de auxílio à projeção e práticas de desenvolvimento de produtos e tecnologias.

O embasamento teórico trazido pela literatura, aliado com conhecimentos da pesquisa exploratória, permite a seleção de atividades, fases, ferramentas e critérios que auxiliem na construção proposição de um modelo e processo de desenvolvimento de produto multidisciplinar.

Já, a aplicação em campo ocorre pela aplicação da proposta no contexto de um projeto de pesquisa e desenvolvimento (P&D), com interação entre iniciativa privada e academia, servindo como forma de avaliação parcial do modelo. Aliado à aplicação, a avaliação do modelo se estabelece de forma qualitativa em pesquisas anônimas com avaliadores provenientes de um contexto empresarial e por especialistas acadêmicos.

1.6 ESTRUTURA DO DOCUMENTO

A estrutura da dissertação é composta por 5 capítulos. O capítulo 1 trata da introdução, onde o tema e o problema de pesquisa são explicitados e justificados, com seus objetivos, justificativas e contribuições esperadas descritos.

No capítulo 2, um referencial teórico é elaborado para apresentar os processos de desenvolvimento das áreas de conhecimento comumente envolvidas em produtos multidisciplinares, assim como a engenharia de sistemas – informações posteriormente utilizadas para elaboração do modelo proposto.

No capítulo 3 é feita a proposição do modelo de desenvolvimento, além do detalhamento de seu processo.

No capítulo 4 o modelo proposto é aplicado em projeto P&D e avaliado de maneira qualitativa conforme os contextos de projeto, empresarial e por especialistas.

Por fim, o capítulo 5 apresenta as conclusões obtidas e propõe recomendações para pesquisas futuras.

2 REFERENCIAL TEÓRICO

O presente capítulo tem como objetivo principal apresentar e organizar informações necessárias para se compreender e embasar os processos de desenvolvimento de produtos conforme diferentes áreas de conhecimento, notadamente a mecânica, eletrônica e software. Essas informações são importantes para o posterior entendimento do modelo e processo propostos.

Inicialmente é abordado o PDP (processo de desenvolvimento de produtos), seu histórico, aplicação e características. Em seguida, se discorre sobre o que a literatura recomenda em termos de procedimentos e boas práticas a respeito do desenvolvimento eletrônico. Da mesma maneira, são abordadas as principais características e métodos encontrados na literatura sobre o desenvolvimento de software. Posteriormente, é tratada a engenharia de sistemas, como uma área de conhecimento para suportar a integração no desenvolvimento de produtos.

Por fim, são apresentadas as considerações finais do capítulo, evidenciando as principais particularidades, diferenças e lacunas na utilização desses processos, modelos e metodologia no desenvolvimento multidisciplinar, bem como quais as possibilidades para que a fronteira desse conhecimento seja ampliada.

2.1 PROCESSO DE DESENVOLVIMENTO DE PRODUTOS

Ainda que haja uma percepção de um caráter aleatório no surgimento de novos produtos, essa aleatoriedade, e o caminho até a disponibilização no mercado, pode ser mais bem direcionada com o uso de um processo estruturado, que engloba desde a definição do problema e geração de ideias, até a concepção e materialização das soluções na forma de produtos, serviços e processos. A essa estruturação se dá o nome de Processo de Desenvolvimento de Produtos (PDP)¹. Esta temática é abordada na literatura com distintos títulos, tais como: metodologia de projeto, metodologia de

¹ Há, dentro da literatura, diversos termos similares – que contêm, entretanto, particularidades – que abrangem a disciplina de desenvolvimento de produtos. Para manter uma uniformidade de escrita, esse estudo adotará o termo Processo de Desenvolvimento de Produtos (PDP).

desenvolvimento de produtos, engenharia do produto, projeto de engenharia e teoria de projeto (Back *et al.*, 2008).

2.1.1 Histórico

Produtos já eram desenvolvidos anteriormente à formalização dos referidos termos acadêmicos. Pode-se considerar que, até a era industrial, o desenvolvimento de produtos era algo mais associado a artes ou artesanato – feitos sob demanda. Posteriormente, os produtos começaram a obter uma característica mais padronizada, os custos foram reduzidos e os produtos começaram a atingir uma gama maior de consumidores (Pahl *et al.*, 2007).

Somente após a Segunda Guerra Mundial, mais especificamente na década de 1960, buscou-se uma maneira explícita de se representar o processo para se desenvolver um produto, com a atividade de projeto sendo tratada como uma disciplina independente (Back *et al.*, 2008). As primeiras observações ocorreram na Europa, em especial na Alemanha, onde a visão do processo de desenvolvimento foi inclusive adotada em norma, a VDI 2221 - *Systematic Approach to the Design of Technical Systems and Products*, a qual sintetiza a filosofia alemã do processo. Nos Estados Unidos esse cenário foi observado, segundo Back *et al.* (2008), a partir de 1985, após a publicação do relatório da American Society of Mechanical Engineering (ASME) intitulado "*Goals and Priorities for Research on Design Theory and Methodology*", que relaciona uma redução da competitividade e qualidade, aliadas a um maior custo de produtos americanos, ao pouco investimento acadêmico e em pesquisas relacionadas a projetos. No Brasil, o estudo de técnicas para sistematizar o desenvolvimento de produtos surgiu academicamente a partir da década de 1980, com destaque para os trabalhos de Back (1983), Rozenfeld *et al.* (2006) e Back *et al.* (2008).

2.1.2 Definições

Para proporcionar uma visão geral sobre o PDP, é primeiro necessário definir o que é um produto. Conforme o dicionário Michaelis, define-se produto como “qualquer coisa fabricada(...) aquilo que é produzido, destinado ao consumo ou comércio” (Produto, 2023). Partindo dessa definição, pode-se

entender que produtos se limitariam a artefatos tangíveis, produzido industrialmente com características e funções, comercializado e usado pelas pessoas ou organizações, de modo a atender a seus desejos ou necessidades (Back *et al.*, 2008).

Uma designação mais abrangente é utilizada por Abreu (1997) e Almeida e Toledo (1992), na qual um produto é aquilo que pode ser oferecido ao mercado para aquisição, utilização ou consumo, podendo incluir objetos físicos (bens), serviços ou ideias. Tal classificação é adotada na presente dissertação, na qual algoritmos, peças mecânicas, softwares ou placas de circuito impresso (PCIs), ou a combinação desses, podem ser considerados produtos. De acordo Jusiuk (2023), há uma distinção entre produtos físicos (ou tangíveis), digitais ou parciais (compostos de ambos). Segundo Guérineau (2022), há uma evolução temporal de produtos unicamente mecânicos, que passaram a ter integração de sistemas eletrônicos e software, dando origem a produtos mecatrônicos, produtos *smart* ou sistemas ciber-físicos. A Figura 1 ilustra produtos em termos de tangibilidade.

Figura 1 - Exemplos de produtos.



Fonte: o autor.

Adota-se, dessa forma, o termo “produto multidisciplinar” aos produtos que incorporam diferentes áreas do conhecimento ou “disciplinas de engenharia” (DE), por exemplo: mecânica, eletrônica, elétrica ou software (Mcharek *et al.*, 2019; Guérineau, 2022). Tais termos são adotados doravante na presente dissertação.

O projeto de um novo produto não significa necessariamente criar um produto original ou disruptivo – novos produtos são comumente desenvolvidos com melhorias, aperfeiçoamentos ou modificações de produtos existentes. A literatura classifica os produtos em (Rozenfeld *et al.*, 2006; Back *et al.*, 2008; Morgan e Liker, 2008):

- Produtos criativos ou revolucionários: são novidades tanto para a empresa, quanto para o mercado. Caracterizam-se por elevados custos e tempo de P&D. Possuem maiores riscos de falha em introdução de mercado, no entanto um potencial de causar grande impacto;
- Produtos aperfeiçoados ou inovativos: partem de produtos já conhecidos na empresa e no mercado. São o resultado de atualizações feitas em produtos existentes para gerar maior valor agregado;
- Produtos de extensão de linha ou variantes: são novidades para o mercado, mas dominados pela empresa. Compostos por reposicionamento de produtos pelo seu uso, formas novas ou versões modificadas de produtos existentes;
- Produtos internalizados: são os produtos já conhecidos no mercado, porém são novos para a empresa. Podem ser concorrentes de produtos revolucionários de outras empresas;
- Produtos localizados: adequação de produto dominado em outros mercados às condições do mercado, fornecedores e processos locais.

O PDP é estabelecido por Pahl *et al.* (2007) como um processo que leva em conta todas as ações do ciclo de vida do produto², com atividades que partem desde a necessidade de mercado ou nova ideia, até a reciclagem ou descarte do produto. Compreende um fluxo de atividades, ferramentas e informações para fomentar e guiar a criatividade e habilidades de projetistas, auxiliando a transformar informações advindas de oportunidades de mercado e tecnológicas em artifícios para a inovação.

² Entende-se por ciclo de vida do produto as etapas sequenciais, desde o planejamento até a retirada do produto do mercado – encerra-se o ciclo de vida quando se finda o compromisso da empresa com o suporte do produto.

2.1.3 Modelos de PDP

Ao se iniciar o desenvolvimento de um produto há uma série de perguntas e decisões que devem ser respondidas (Krishnan e Ulrich, 2001):

- Qual será o valor alvo dos atributos do produto?
- Qual será o conceito do produto?
- Que variações do produto serão ofertadas?
- Como é a arquitetura do produto?
- Qual será a forma física geral e desenho industrial do produto?
- Quais componentes serão projetados especificamente para o produto?
- Quem irá projetar e produzir o produto?
- Qual é a configuração da cadeia de fornecimento física?
- Que tipo de processo será utilizado para montar o produto?
- Quem irá desenvolver e fornecer o equipamento para o processo?

Para endereçar essas indagações, o PDP é comumente dividido em fases sequenciais, onde a cada mudança de fase um modelo mais detalhado substitui um anterior, mais abstrato. As fases são constituídas de atividades e tarefas, cuja execução visa gerar um conjunto de entregas, que avançam e estabelecem um novo patamar evolutivo do produto (Rozenfeld *et al.*, 2006; Tomazelli, 2006; Pahl *et al.*, 2007; Back *et al.*, 2008).

A determinação das fases e seus marcos evolutivos são, no entanto, um processo mutável e contínuo na literatura, tanto em seu detalhamento, quanto na configuração. Diversos autores propuseram alternativas de modelo de PDP. Um dos autores pioneiros no assunto, Asimov (1968) sugere que as principais fases de desenvolvimento de produto são (Figura 2):

- Estudo de Viabilidade;
- Projeto Preliminar;
- Projeto Detalhado;
- Planejamento para Manufatura;
- Planejamento para Distribuição;
- Planejamento para Consumo;
- Planejamento para Retirada.

Figura 2 - Fases do PDP segundo Asimov.



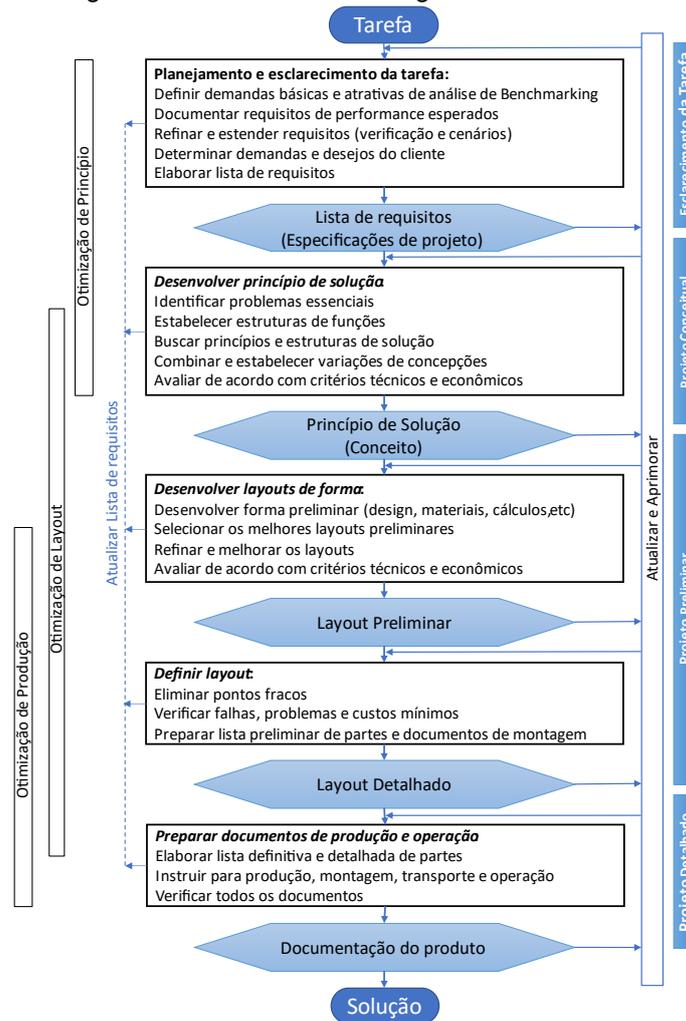
Fonte: adaptado de Asimov (1968).

Na década de 1970, Pahl e Beitz propuseram um modelo com as seguintes fases (Figura 3):

- Planejamento e esclarecimento da tarefa: especificação da informação;
- Projeto conceitual: especificação do princípio de solução (conceito);
- Projeto preliminar: especificação do *layout* (construção);
- Projeto detalhado: especificação da produção³.

³ A obra consultada de Pahl et al. (2007) é a edição em inglês. Os termos citados no texto e figura são de tradução do autor e podem divergir da edição em português.

Figura 3 – Modelo de PDP segundo Pahl e Beitz.

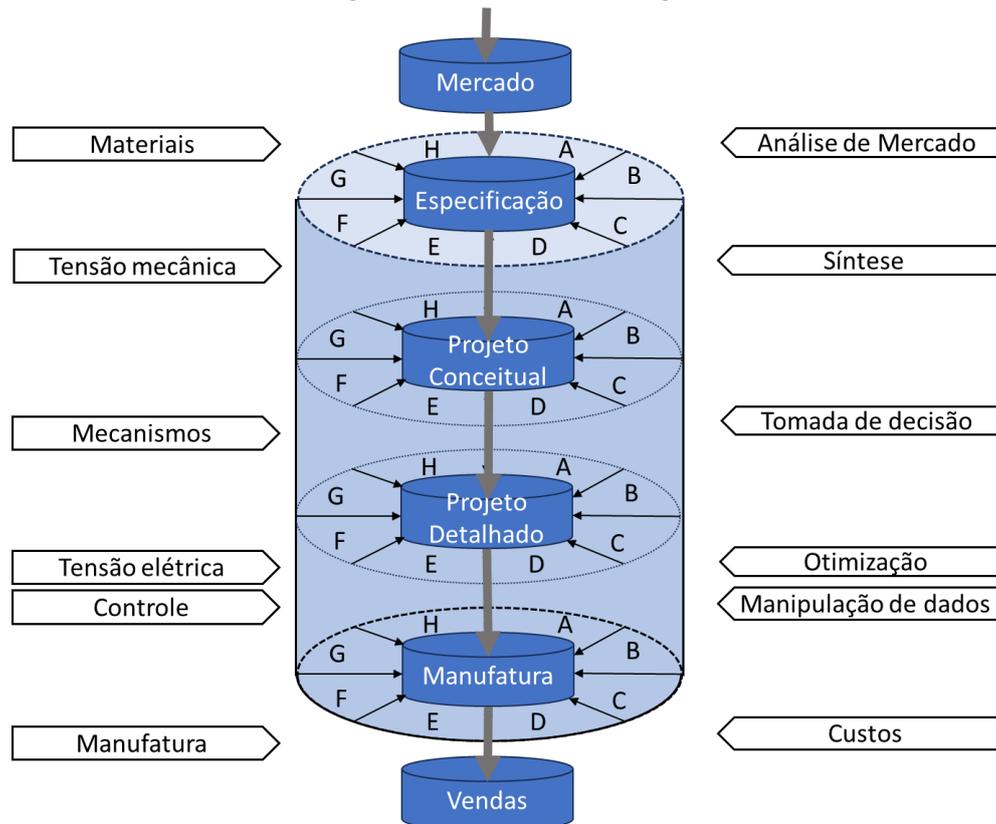


Fonte: adaptado de Pahl *et al.* (2007).

O modelo proposto por Pugh (1991, *apud* Reinert, 2017), conhecido como “*Total design*” (Figura 4), traz 6 fases principais:

- Mercado: análise das necessidades do usuário;
- Especificação: esclarecimento de quais necessidades serão atendidas pelo produto;
- Projeto Conceitual: especificação clara da abordagem para gerar o conceito;
- Projeto Detalhado: detalhamento técnico do conceito para tornar-se um produto real;
- Manufatura: planejamento e fabricação dos primeiros modelos;
- Vendas: colocar o produto no mercado.

Figura 4 - Modelo Total Design.

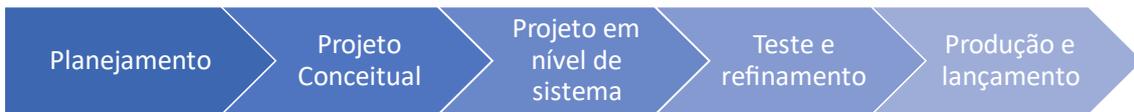


Fonte: adaptado de Pugh (1991) apud Reinert (2017).

A proposta de Ulrich e Eppinger (1995, *apud* Reinert, 2017), tem uma característica linear, como mostra a Figura 5, na qual os autores dividem o PDP em seis fases principais:

- Planejamento: declaração da missão do projeto, mercado-alvo, premissas e restrições;
- Projeto conceitual: geração e avaliação de conceitos para o produto;
- Projeto em nível de sistema: definição da arquitetura do produto e decomposição em componentes e subsistemas;
- Projeto detalhado: definição final de geometria, materiais e tolerâncias;
- Teste e refinamento: construção e avaliação de protótipos;
- Produção e lançamento: fabricação de primeiras peças do produto a fim de definir métodos e processos de produção.

Figura 5 - Modelo de Ulrich e Eppinger.



Fonte: Adaptado de Ulrich e Eppinger (1995) apud Reinert (2007).

O modelo de referência de Rozenfeld *et al.* (2006) representa o PDP em três macrofases (Pré-desenvolvimento, Desenvolvimento e Pós-desenvolvimento) e oito fases. Uma das características desse modelo é a utilização de *gates* formais a cada mudança de fase, conforme Figura 6:

- Planejamento do projeto: determinação do escopo do projeto;
- Projeto Informacional: especificação de requisitos com valores meta (início do desenvolvimento);
- Projeto Conceitual: definição da arquitetura do produto;
- Projeto Detalhado: especificação final do produto;
- Preparação da Produção: liberação da produção do produto;
- Lançamento do Produto: lançamento do produto ao cliente (fim da atividade de desenvolvimento);
- Acompanhar produto/processo: acompanhamento do produto no mercado e na empresa;
- Descontinuar produto: final do ciclo de vida do PDP.

Figura 6 - Modelo de Rozenfeld *et al.*

Fonte: Adaptado de Rozenfeld *et al.* (2006).

Voltado à realidade brasileira, especialmente para a concepção de máquinas agrícolas, Romano (2003) propôs um modelo de referência – ilustrado na Figura 7. Tal modelo foi expandido por Back *et al.* (2008) como o Processo de Desenvolvimento Integrado de Produtos (PRODIP), baseado em 3 macrofases (planejamento, elaboração de projeto do produto e implementação do lote inicial) e 8 fases, cujo resultado deve ser avaliado para permitir avançar à fase seguinte. As fases do modelo PRODIP são:

- Planejamento do projeto: detecção da oportunidade de negócio;
- Projeto Informacional: estabelecimento das especificações de projeto;
- Projeto Conceitual: geração de conceito de produto;
- Projeto Preliminar: definição da configuração do produto;
- Projeto Detalhado: documentação detalhada do produto e processo de manufatura;
- Preparação da Produção: produção de lote piloto para validar conformidade de produto e processo;
- Lançamento: produção de lote inicial do produto;
- Validação: validar produto junto ao cliente e desmobilizar a equipe.

Figura 7 - Modelo PRODIP.



Fonte: adaptado de Back *et al.* (2008).

Outra abordagem a ser citada, apesar de não se tratar de um modelo baseado em fases propriamente ditas, é o “Desenvolvimento de Produtos Lean (Lean PDP)”. Morgan e Liker (2008) indicam algumas recomendações sobre a aplicação do Lean PDP:

- Definir um "espaço do projeto" ou requisitos do sistema;

- Criar múltiplas alternativas (ou soluções) de projeto e processo baseadas em padrões (inclusive seções comuns de construção);
- Testes rápidos e objetivos do programa, analisando o impacto de cada alternativa sobre o custo, a qualidade e o desempenho;
- Concentrar-se rigorosamente nas características essenciais de cada alternativa;
- Combinar as características comuns a todas as alternativas;
- Focar energia e esforços visando uma solução única de projeto e processo.

Ao analisar as diferentes abordagens dos modelos de PDP, é possível inferir que os modelos, de uma forma geral, compreendem três etapas principais do ciclo de vida do produto:

1. Pré-desenvolvimento, planejamento ou análise de mercado;
2. Desenvolvimento, ou geração de soluções/produtos;
3. Pós-desenvolvimento, implementação ou produção.

A etapa de desenvolvimento, é aquela na qual são tomadas decisões que podem ou não viabilizar o novo produto, onde são definidas as principais soluções construtivas e, para isso, se empenha esforço e investimento em engenharia. Como essa etapa é o enfoque da presente dissertação, suas principais entregas, atividades, ferramentas e recomendações são detalhadas a seguir, de acordo com quatro fases propostas por Romano (2003) e Back *et al.* (2008): Projeto Informacional, Projeto Conceitual, Projeto Preliminar e Projeto Detalhado.

2.1.3.1 *Projeto Informacional*

O projeto informacional é a primeira fase da etapa de desenvolvimento propriamente dito. Durante essa fase, se cria, a partir do planejamento do produto e de necessidades de clientes, a lista de requisitos de produto e projeto, bem como as especificações – informações qualitativas e quantitativas que irão direcionar os esforços e próximos passos de desenvolvimento (Rozenfeld *et al.*, 2006; Tomazelli, 2006; Pahl *et al.*, 2007).

Para o correto estabelecimento dessa lista, é necessário um embasamento sobre o problema, cliente, tecnologia disponível e objetivos. O

primeiro passo é obter um entendimento claro e completo do contexto, ou seja, do problema a ser solucionado. Deve-se examinar com maior profundidade informações tais como: tipo de produto, mercado do produto, tecnologias disponíveis, desejos do cliente, objetivos e restrições do produto. A primeira fonte para o esclarecimento são o contato com o cliente e o planejamento do produto. O entendimento é refinado através da busca do estado da arte da tecnologia em fontes principais, tais como (Rozenfeld *et al.*, 2006; Tomazelli, 2006; Pahl *et al.*, 2007):

- Produtos similares: através de catálogos, manuais e *datasheets*;
- Patentes: busca em bancos disponíveis de patentes de produtos similares e/ou com funções correlacionadas;
- Fontes acadêmicas: busca em dicionários técnicos, livros, revistas científicas, anais de congressos, entre outros;
- Centros de produção: no caso de reprojeto de produtos existentes.
- Normas técnicas reguladoras ou legislações vigentes: fonte de informações, especialmente no que diz respeito a objetivos e restrições.

Entende-se que há diferentes partes interessadas (*stakeholders*) envolvidas em diferentes fases do produto, portanto há distintas demandas a serem cumpridas. Para Pahl *et al.* (2007), requisitos e especificações podem ser considerados como as informações mais importantes e relevantes para um projeto. Uma das ferramentas mais conhecidas para seu estabelecimento é a QFD (*Quality Function Deployment*). Essa ferramenta permite, segundo Back *et al.* (2008), documentar e visualizar as necessidades dos clientes, as quais irão ser transformadas em requisitos de usuário e requisitos de projeto, que posteriormente são priorizados por meio de uma comparação cruzada com produtos disponíveis no mercado.

Os requisitos que o produto deverá cumprir, associados a parâmetros quantitativos e mensuráveis, definem as especificações do produto. As especificações promovem (Krishnan e Ulrich, 2001; Pahl *et al.*, 2007):

- Entendimento do cliente;
- Entendimento do produto;
- Limitação no escopo do produto;
- Representação do produto como um vetor de atributos;

- Direcionamento para a geração e/ou seleção de soluções;
- Embasamento para critérios de avaliação;
- Documentação do projeto.

Conforme o avanço do projeto, quaisquer mudanças ou adições ao objetivo original devem gerar mudanças na lista de especificações, atualizando-a e refletindo o progresso do desenvolvimento.

2.1.3.2 *Projeto Conceitual*

O projeto conceitual é a fase em que o produto começa a tomar forma. As atividades da equipe nessa fase envolvem a criação, representação e seleção de soluções para o problema, ou seja, para atender a função global que o produto deve ser capaz de executar. Portanto, as decisões tomadas nessa fase influenciam de forma substancial os resultados atingidos nas próximas fases (Pahl *et al.*, 2007; Back *et al.*, 2008).

Para o estabelecimento da concepção, Pahl *et al.* (2007) sugerem um passo a passo para gerar, selecionar e otimizar soluções com menor esforço, sendo as ações:

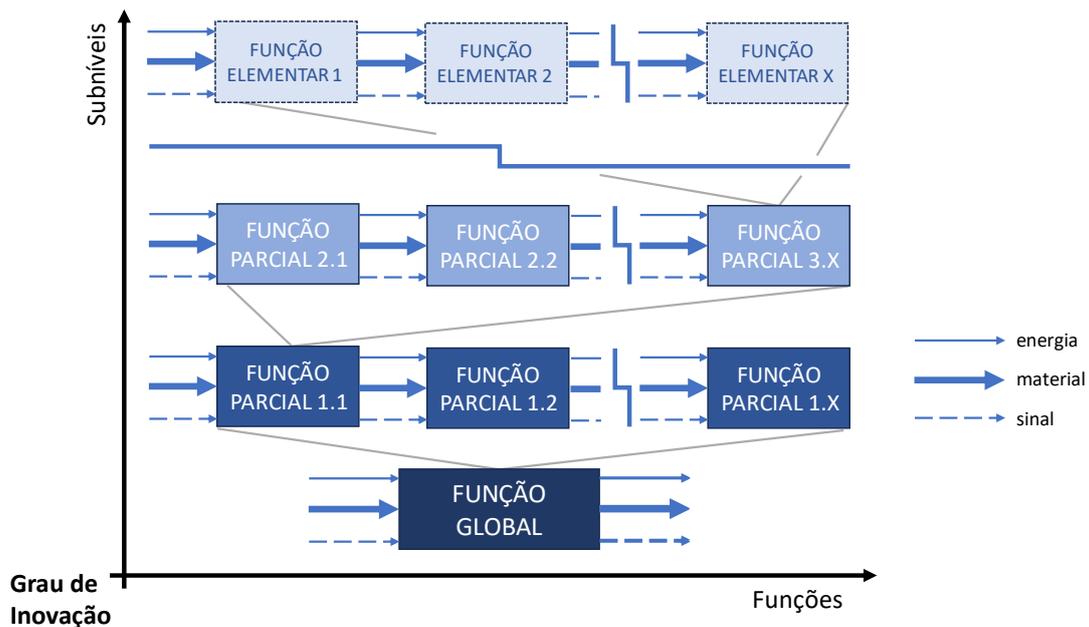
1. Identificar problemas essenciais através de abstração;
2. Estabelecer estruturas de funções;
3. Buscar princípios e estruturas de solução;
4. Combinar e estabelecer variações de concepções;
5. Avaliar e selecionar concepção.

Em linhas gerais, a fase de projeto conceitual pode ser definida com um momento de análise (passos 1 e 2), que busca abstrair o problema, e um momento de síntese (passos 3 a 5), que busca concretizar ideias.

Uma ferramenta que auxilia na abstração é a síntese funcional, na qual o problema é descrito conforme os fluxos de energia, material e sinal necessários ou estimados para a execução da ação (função) que o produto deve exercer. Inicialmente, por meio das especificações do produto, define-se a função global (verbo e substantivo), ou seja, qual o principal objetivo que o produto deve cumprir. Essa pode, então, ser desmembrada em subfunções de menor complexidade, estabelecendo-se uma estrutura de funções. Procede-se com a decomposição de funções até se alcançar um nível de complexidade tal que

permita buscar e se identificar princípios de solução. O número de níveis de desdobramento, ou de funções por nível, é determinado essencialmente pela complexidade e novidade do problema (Figura 8). As funções no último nível de desdobramento são identificadas como funções elementares (Pahl *et al.*, 2007; Back *et al.*, 2008).

Figura 8 - Estrutura funcional de acordo com grau de inovação.



Fonte: o autor.

Com a estrutura funcional definida, para cada função elementar deve-se buscar os efeitos físicos envolvidos e princípios de solução capazes de realizar ou executar seu objetivo, que, associados, geram uma concepção. Uma ferramenta proposta para guiar a geração de concepções nesse estágio é a Matriz Morfológica (ou Análise Morfológica), que auxilia sistematicamente na combinação de soluções. Segundo Back *et al.* (2008), a Matriz Morfológica “consiste em uma pesquisa sistemática de diferentes combinações de elementos ou parâmetros, com o objetivo de encontrar uma nova solução para o problema”. Em sua utilização, as funções elementares encontradas são inseridas uma em cada linha de uma matriz. Alternativas de princípios de solução — encontradas em catálogos, pesquisas de literatura ou patentes, métodos intuitivos ou experiência — são dispostos nas colunas, de forma gráfica ou textual, conforme observado na Figura 9. Ao associar sistematicamente um

princípio de solução que satisfaça uma função de cada linha, um conceito de solução é gerado (Pahl *et al.*, 2007).

Figura 9 - Matriz Morfológica.

	PS1	PS2	...	PSx
Função Elementar 1				
Função Elementar 2				
...				
Função Elementar x				

Fonte: o autor.

As propostas de solução elaboradas necessitam, então, ser avaliadas. A tomada de decisão pode ser feita através de um processo que envolve a comparação de conceitos e sua valoração conforme o cumprimento potencial de objetivos. Pugh (1991, *apud* Back *et al.*, 2008) propôs o método da Matriz de Decisão (ou Matriz de Pugh) na qual as alternativas de solução são dispostas nas colunas e os critérios de avaliação nas linhas, uma das alternativas deve ser escolhida como referência com a qual as demais são comparadas em cada critério. Os critérios de avaliação podem ser baseados nas especificações de projeto e/ou podem ser critérios gerais relacionados a custos, segurança, usabilidade, riscos, qualidade ergonomia, entre outros. Ao final da aplicação da Matriz de Decisão deve ser selecionada uma concepção que seguirá no processo de desenvolvimento. Essa concepção compreende uma descrição aproximada das tecnologias, princípios de funcionamento e formas, e pode ser representada por meio da estrutura de funções, diagramas (de circuito ou fluxo), esboços ou modelo tridimensional que, frequentemente, são acompanhados por uma explicação textual (Krishnan e Ulrich, 2001; Rozenfeld *et al.*, 2006; Pahl *et al.*, 2007; Back *et al.*, 2008).

Apesar da quantidade de tarefas realizadas nessa fase, o tempo despendido pode ser compensado nas fases subsequentes, pois retrabalho

pode ser evitado. Durante as fases de Projeto Preliminar e Detalhado se torna mais complexo e custoso corrigir deficiências fundamentais vindas de uma má escolha de concepção (Pahl *et al.*, 2007).

2.1.3.3 *Projeto Preliminar*

O projeto preliminar é a fase em que deve ser estabelecido o *layout* final do produto e o estudo de viabilidade técnica e econômica. Atividades relacionadas ao dimensionamento, simulações e prototipagens são executadas, de forma a se aprofundar no detalhamento da solução, aproximando-a de um produto capaz de cumprir as funções requeridas (Back *et al.*, 2008).

Nessa fase são realizados os primeiros desenhos formais do produto e a utilização deliberada de engenharia e cálculos para que, ao final, se estabeleçam (Pahl *et al.*, 2007; Back *et al.*, 2008):

- Dimensionamento do produto;
- *Layout* de forma e materiais;
- Definição de sistemas, subsistemas e componentes;
- Estabelecimento de soluções para toda e qualquer função do produto;
- Alinhamento de processos de produção;
- Viabilidade técnica do projeto.

A fase é caracterizada pela consolidação das ideias em ciclos repetidos de deliberação e verificação. Algumas características do conceito escolhido podem, nessa fase, ser identificadas como inadequadas. Nesses casos, recomenda-se reexaminar as decisões realizadas na fase anterior (projeto conceitual) pois, mesmo sendo corretamente executado o projeto preliminar, corrigir um conceito inadequado torna-se custoso e adiciona riscos ao desenvolvimento.

Pahl *et al.* (2007), propõem um passo a passo para a execução das atividades da presente fase, partindo da concepção:

1. Identificar as especificações que têm uma relevância crucial na definição de forma;
2. Identificar limitações espaciais que determinam ou restringem a forma;

3. Produzir um *layout* inicial com ênfase no estabelecimento dos portadores das funções principais, ou seja, os componentes e montagens que atuam nas funções principais;
4. Desenvolver um *layout* preliminar em escala e definição de formas para os portadores das funções principais;
5. Desenvolver *layouts* preliminares para os portadores de funções remanescentes;
6. Determinar quais funções auxiliares são necessárias e, quando possível, explorar soluções conhecidas;
7. Produzir um *layout* detalhado para os portadores de função principais de acordo com padrões, regulamentações, cálculos detalhados e descobertas experimentais;
8. Desenvolver *layout* detalhado para os portadores de funções auxiliares, adicionando peças padronizadas e comerciais;
9. Avaliar o *layout* em critérios técnicos e econômicos;
10. Ajustar e corrigir *layout* preliminar geral;
11. Otimizar e completar *layout* ao eliminar pontos fracos identificados durante a avaliação;
12. Buscar no projeto problemas no cumprimento de funções, compatibilidade espacial e outros fatores de problemas;
13. Preparar uma lista preliminar de peças, assim como documentos preliminares de produção e montagem;
14. Finalizar o *layout* detalhado e prosseguir para a fase de projeto detalhado.

Nota-se, nesse procedimento sugerido, a existência de dois *layouts* distintos, um *layout* preliminar (passos 1 a 9) e um *layout* detalhado (passos 10 a 14). Nesse processo, diversos detalhes devem ser esclarecidos, confirmados e otimizados, como a decisão sobre quais componentes devem ser comprados e quais componentes precisam ser projetados (a decisão “*make or buy?*”, traduzida livremente como “construir ou comprar?”). Enquanto decisões de detalhamento de projeto são realizadas e refinadas, a solução pode ser prototipada para avaliação e adequação, verificação de função, fabricação e

testes específicos, ou seja, é proposto um ciclo iterativo de Projetar-Construir-Testar-Otimizar (Krishnan e Ulrich, 2001; Pahl *et al.*, 2007).

Para a avaliação do projeto preliminar, Pahl *et al.* (2007) indicam alguns aspectos técnicos essenciais para uma solução adequada (Pahl *et al.*, 2007):

- Clareza: a não ambiguidade nas funções de um projeto facilita a predição da performance do produto e em muitos casos poupa tempo e análises de custo;
- Simplicidade: geralmente garante viabilidade econômica. Um número menor de componentes e formas simples são produzidos de maneira mais rápida e fácil.
- Segurança: impõe uma abordagem consistente para problemas de robustez, confiabilidade, prevenção de acidentes e proteção do meio ambiente.

Uma vez que o *layout* seja aprovado conforme os critérios, pode-se seguir para a última fase do desenvolvimento, o Projeto Detalhado.

2.1.3.4 *Projeto Detalhado*

O Projeto Detalhado encerra a etapa de desenvolvimento do produto. Essa fase apresenta os seguintes propósitos: aprovação do protótipo, finalização das especificações do produto e seus componentes, detalhamento do plano de manufatura (Back *et al.*, 2008).

Há dois momentos distintos que ocorrem na fase: um momento final de criação, no qual propriedades sobre arranjo, formas, dimensões e superfície de todas as partes individuais são finamente definidas e os materiais especificados; e um momento de verificação, com meios de produção minuciosamente avaliados, custos estimados, e todos os desenhos e outros documentos de produção produzidos.

A fase de Projeto Detalhado resulta nas especificações finais do produto, na forma de uma gama de documentos, incluindo desenhos técnicos detalhados e listas de materiais, assim como instruções para produção (Pahl *et al.*, 2007).

Apesar de não haver ferramentas claras na literatura para essa fase, com as saídas esperadas bem-definidas. Pahl *et al.* (2007) propõem os seguintes passos:

1. Finalizar o *layout* definitivo, compreendendo o desenho detalhado de componentes, otimização de formas, materiais, superfícies, tolerâncias e ajustes, com atenção voltada à produção ou padrões de mercado;
2. Integrar componentes individuais em montagens e ao produto final com o auxílio documental de desenhos técnicos, lista de peças e identificação de sistemas;
3. Finalizar documentos de produção com as instruções de produção, montagem, transporte e operação.

Uma vez encerrada essa fase, encerra-se o desenvolvimento do produto propriamente dito. Durante o pós-desenvolvimento pode-se ressaltar como atividades importantes: a certificação do produto, a homologação de produção e o acompanhamento do produto no mercado (Rozenfeld *et al.*, 2006; Pahl *et al.*, 2007).

2.1.4 Considerações sobre o Processo de Desenvolvimento de Produtos

As práticas listadas nessa seção são bastante difundidas para produtos da indústria metal mecânica, como automóveis, aeronaves, máquinas-ferramenta (Rozenfeld *et al.*, 2006).

Muitas das atividades apresentadas têm aspectos voltados à engenharia mecânica, tais como definir *layout* de forma e materiais, especificar tolerâncias, desenvolver plano de manufatura. Porém, pode-se questionar se, com a revolução digital da contemporânea (quarta Revolução Industrial), os modelos apresentados continuam adequados, especialmente para produtos multidisciplinares, ou que sejam em parte ou integralmente não tangíveis, tais como produtos *smart* ou sistemas ciber-físicos.

As próximas seções apresentam as práticas encontradas na literatura para o desenvolvimento de soluções eletrônicas e de software, essenciais para o desenvolvimento de produtos multidisciplinares.

2.2 DESENVOLVIMENTO ELETRÔNICO

Produtos eletrônicos fazem parte da rotina da grande maioria da população, desde simples calculadoras de bolso até os onipresentes *smartphones*. O que caracteriza esses produtos é a existência de um circuito eletrônico embarcado, que consiste numa Placa de Circuito Impresso (PCI).

Para Rigo (2019), as PCIs “são componentes fundamentais na produção de todos os equipamentos eletrônicos e constituem um complexo trabalho de engenharia, pois envolvem diversos objetivos conflitantes de várias áreas do conhecimento”. Ademais, o projeto dessas placas tem um caráter constituinte, ou seja, deve ser realizado de forma que possibilite a integração do mesmo como parte de um produto, no qual toda PCI deverá atender requisitos de integração mecânica e com o software embarcado. Tais fatores que podem gerar a necessidade de diversas iterações de projeto para viabilizar as acomodações e ajustes necessários para a aplicação (Danković *et al.*, 2013).

As subseções que seguem abordam de forma mais detalhada o que constitui uma PCI e o processo de desenvolvimento eletrônico em si.

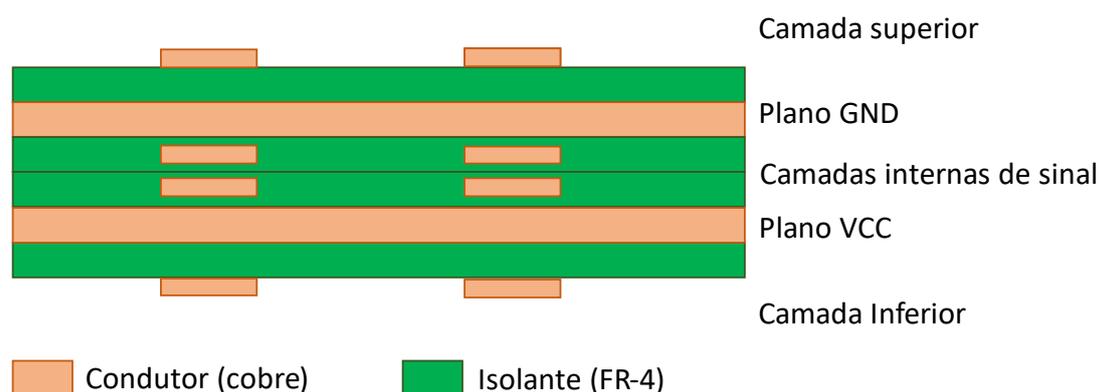
2.2.1 Definições

A PCI é o elemento principal de quase todo projeto eletrônico. É a placa que contém os componentes e conexões elétricas de maneira compacta de eficiente (Rigo, 2019; Russ, 2022).

O principal propósito da PCI é apresentar as ligações necessárias para conectar os componentes de um circuito eletrônico, além de prover estrutura e rigidez mecânica para, por exemplo, acomodar conectores. De forma a substituir circuitos com cabos condutores, uma PCI é fabricada por meio de um processo de gravação, que estabelece as conexões necessárias através de trilhas, que consistem em caminhos condutores com objetivo de transmitir energia (e/ou informação) em formas de ondas eletromagnéticas (Rigo, 2019). Essas trilhas são gravadas de forma compacta, o que favorece propriedades físicas como a integridade de sinal e redução de dimensões físicas, e são constituídas de cobre disposto em uma placa com camada de um material isolante, normalmente a fibra de vidro FR-4 (*Flame Retardant class 4*), tipicamente verde (Russ, 2022).

A depender da complexidade e do número de circuitos, por vezes não se tem espaço suficiente para realizar todas as trilhas em somente uma camada da placa, visto que as trilhas são projetadas com espaçamento e largura mínimos por restrições de fabricação e resistência elétrica. Dessa maneira, há conexão de trilhas entre camadas diferentes por estruturas verticais chamadas de vias. Segundo Russ (2022), normalmente as PCIs são formadas por ao menos duas camadas de trilhas. De uma maneira geral, há uma relação direta entre o número de camadas de uma PCI, seu custo e funcionalidades que a placa que é capaz de desempenhar – na maioria dos produtos eletrônicos para consumo se encontram PCIs de quatro camadas. A Figura 10 ilustra em corte a configuração típica de uma PCI multicamadas. Além da placa base, a PCI s é formada pelos componentes posicionados no *footprint* de componente, inseridos e soldados a ela.

Figura 10 - Configuração típica de uma PCI.



Fonte: adaptado de Russ (2022).

2.2.2 Processo de desenvolvimento

Na literatura identificam-se abordagens mais voltadas às boas práticas na projeção⁴ de PCIs do que a um método ou modelo propriamente dito, capaz de guiar o projetista ao longo do processo de desenvolvimento do eletrônico, contabilizando seu ciclo de vida. Ainda assim, o desenvolvimento de PCIs comumente enfrenta restrições de tempo, tendo em vista que,

⁴ O termo “projeção” foi proposto por Romano (2003) como uma macro-fase de desenvolvimento. Essa dissertação adota o termo como uma ação de projetar, tradução do termo inglês “*design*”.

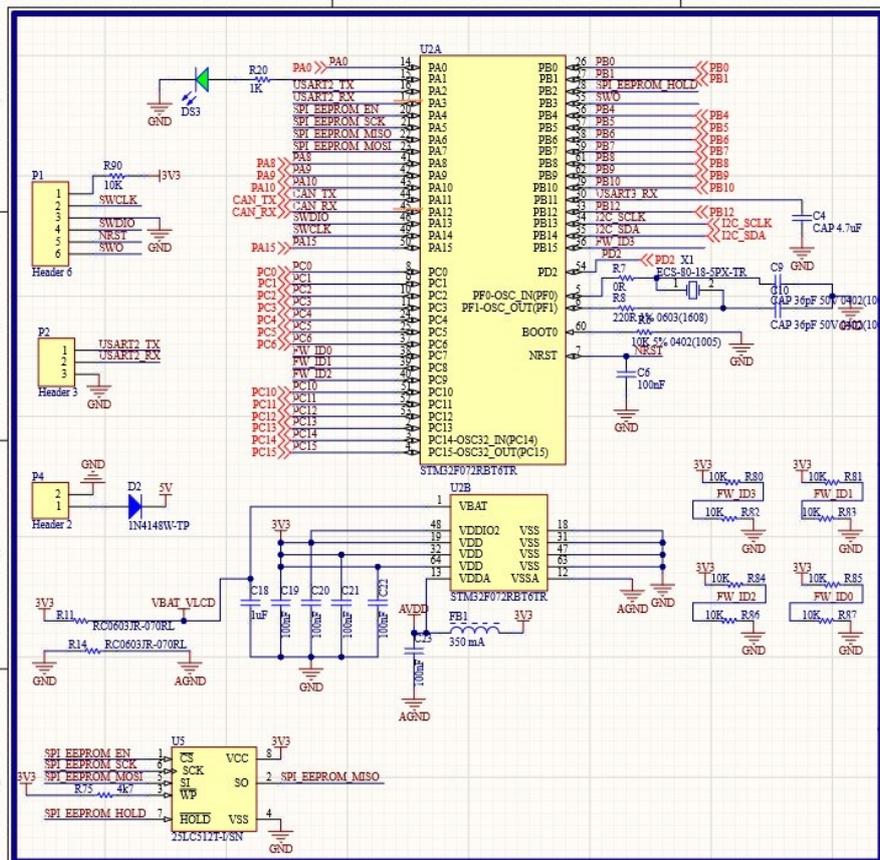
frequentemente, está associada a projetos de sistemas cada vez mais complexos, compactos e processamentos mais velozes (Rigo, 2019).

Uma abordagem convencional no projeto de PCIs é concentrar-se na disponibilidade tecnológica e de produção, na escolha de componentes, *layout* da placa, suas trilhas e arranjos físicos, interconexões externas, regras de projeto e roteamento. Essas definições estão vinculadas e são dependentes das funções às quais a PCI deve cumprir e onde ela será instalada, ou seja, de acordo com os requisitos de montagem mecânica e do produto (Danković *et al.*, 2013; Russ, 2022).

O primeiro passo para o desenvolvimento de uma PCI, de acordo com Russ (2022), é a escolha de componentes principais como o processador e os circuitos especializados (*e.g.* FPGAs, memórias, *displays*, circuitos analógicos ou fontes de potência) para realização da função principal. O autor ressalta que essa escolha é considerada uma forma de arte, pois são confrontadas características e funções a custos, tamanho e consumo de energia, além da disponibilidade de componentes no mercado. A seguir, são selecionados os componentes passivos, de menor valor agregado, como resistores, capacitores, diodos, entre outros. Essa escolha é mais simples, pois esses componentes são normalmente orientados por recomendações advindas de informações técnicas dos componentes principais. A seleção de componentes se encerra com a criação de um conjunto de informações para cada componente: um símbolo esquemático com as conexões de cada pino do componente e um *footprint*, *ou seja*, o arranjo físico do componente na PCI. Nesse sentido, deve haver uma consistência nessas representações, sobretudo no mapeamento de pinos.

Uma vez que todos os componentes são selecionados, o projetista cria um esquemático, que é um diagrama que mostra todos os componentes e suas conexões, ilustrado na Figura 11. Esse passo pode ser considerado o mais importante no projeto da PCI, pois especifica a topologia do projeto, informações de valores dos componentes e todos os aspectos das conexões. Frequentemente, fabricantes dos componentes principais produzem um projeto de referência com as devidas conexões, recomendações de componentes e fórmulas para calcular valores de componentes passivos (Russ, 2022).

Figura 11 - Esquemático de uma PCI.



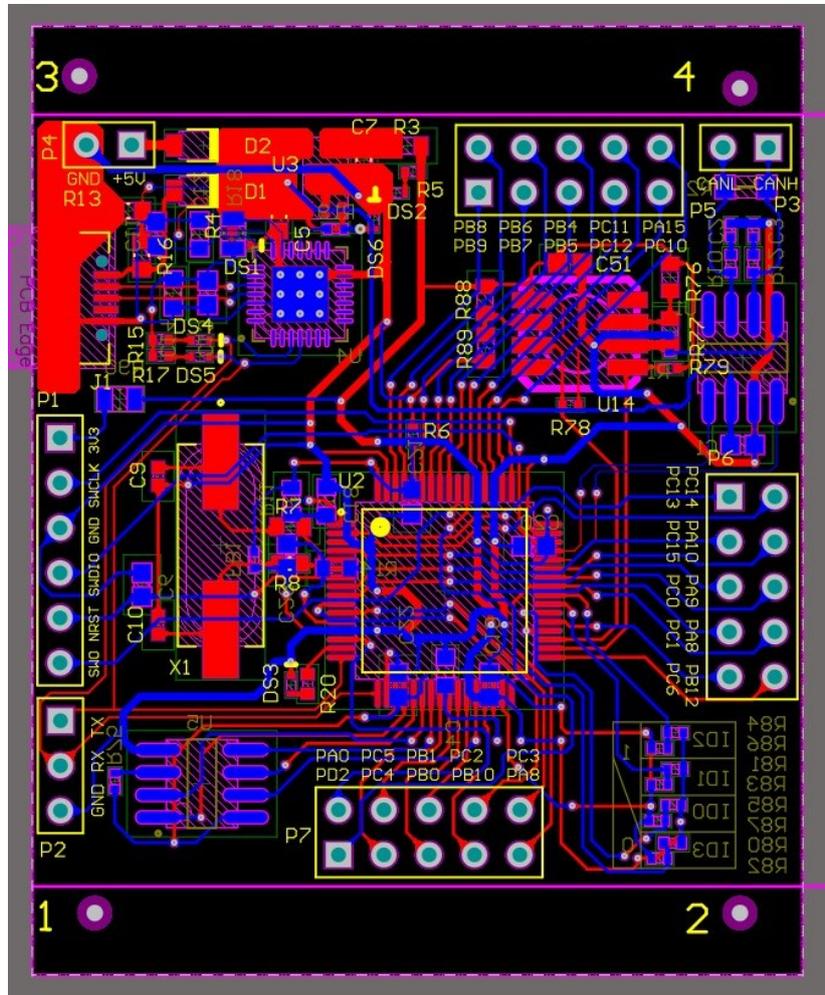
Fonte: o autor.

Do esquemático é gerado um *layout*, que é onde o projetista expressa as informações do tipo e qual a função de cada camada, bem como quais furos necessitam ser feitos para as vias. No *layout* também são especificados os *footprints* que direcionam como cada componente estará colocado na placa, processo conhecido como *placement*. Com a especificação dos *footprints*, pode se iniciar o roteamento de trilhas, ou seja, determinar as rotas para conexão dos componentes. Para Rigo (2019), esse passo pode exigir um grande esforço em estratégias, cálculos, ponderações e compensações, tendo em vista que há, muitas vezes, restrições de largura e espaçamento de acordo com fornecedor.

Outro dado que depende de informações de projeto e fornecedor é o *stack up* (especificação da espessura do isolante e condutor), que busca uma impedância característica dentro das espessuras ofertadas pelo fabricante. Por fim, no *layout* também se adicionam as marcações de serigrafia, tais como avisos de alta tensão, logos empresariais, nomes dos projetistas, local para

posicionamento e polaridade de componentes (Russ, 2022). A Figura 12 mostra o *layout* de uma placa de circuito impresso.

Figura 12 - Layout de uma PCI.



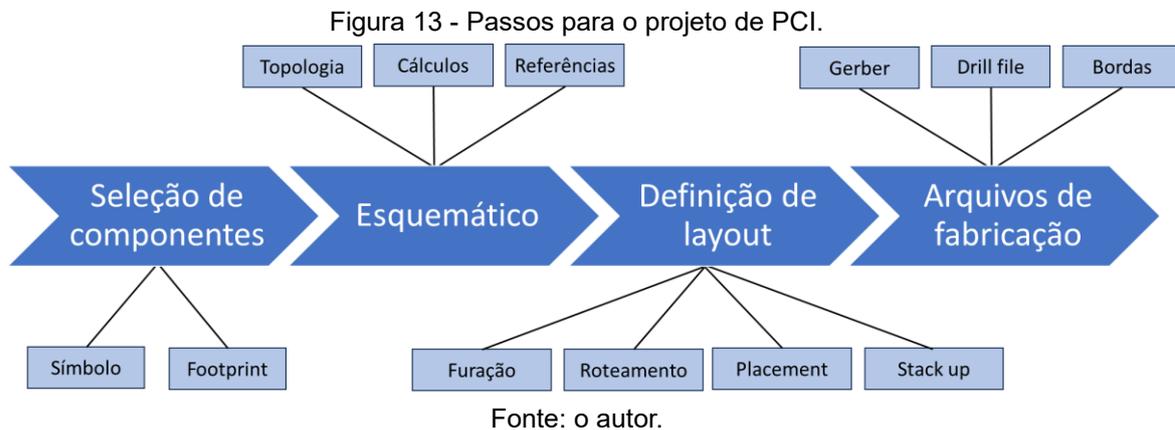
Fonte: o autor.

Com o *layout* terminado, a placa está pronta para fabricação. O resultado entregável de um projeto completo de PCI é uma série de arquivos de projeto prontos para ser enviados ao fabricante para realizar a manufatura da placa base e, posteriormente, para o montador, que adicionará os componentes nas placas. Tais arquivos podem ser descritos como (Rigo, 2019):

- Arquivos Gerber: descrevem as gravações na placa, como camadas de cobre, máscara de solda, serigrafia;
- Relatório de perfuração: detalhamento de ferramentas, tamanhos de furos, contagem de furos e especificações para as vias;

- Borda da placa: descrição de forma e tamanho da borda da placa e pontos de recorte.

A Figura 13 resume os principais passos descritos na literatura para desenvolvimento de uma placa de circuito impresso.



Algumas ferramentas que podem auxiliar no projeto da PCI são os softwares EDA (*Electronic Design Automation*), os quais englobam ferramentas CAE (*Computer Aided Engineering*) e CAD (*Computer Aided Design*). A primeira ferramenta auxilia no desenho esquemático, permitindo criar e gerenciar componentes, realizar representações lógicas, análises de integridade de sinais, simulação de circuito e até mesmo gerar lista de compras. Já, a segunda compreende o aspecto físico do projeto, com modelagem 3D da placa, camadas e componentes, roteamento de trilhas, posicionamento automático de componentes, análises térmica e de integridade de sinais, geração dos arquivos para fabricação (Rigo, 2019).

Além das ferramentas, há algumas recomendações da literatura que ressaltam aspectos importantes no projeto de PCIs, conforme o Anexo A.

2.2.3 Considerações sobre o Desenvolvimento Eletrônico

Há uma natureza distinta no desenvolvimento de PCIs quando se compara àquele para equipamentos mecânicos, sobretudo no que tange às atividades de geração de conceitos: o desenvolvimento eletrônico é inicialmente pautado na seleção e escolha de componentes comerciais, os quais irão direcionar boa parte do conceito gerado. Apesar da importância cotidiana, o

desenvolvimento eletrônico não é tratado de forma específica conforme fundamentos tradicionais do PDP, que são enraizados especialmente em conhecimentos e perspectivas advindos da engenharia mecânica (Romeral *et al.*, 2023).

Um estudo conduzido no processo de desenvolvimento de produtos eletroeletrônicos em empresas incubadas observou a não utilização de modelos de referência durante o desenvolvimento de placas eletrônicas. Os autores Pagan *et al.* (2011) ressaltam que, “apesar de realizarem algumas das atividades recorrentes da fase de projeto conceitual, nenhuma das empresas segue um modelo de referência”. A principal ferramenta observada no estudo é somente durante a definição da arquitetura da PCI, com utilização de DFM (*Design for Manufacturing*, traduzido como Projeto para manufatura).

Ainda que o desenvolvimento de PCIs possua uma lógica de desenvolvimento mapeada, ele se difere de algumas das atividades sugeridas em modelos de referência para o PDP, como: não se observa na literatura um acompanhamento no ciclo de vida de produtos eletrônicos; não se identificam ferramentas que guiem ou fomentem a criatividade e habilidade dos projetistas; não há uso de abstrações para chegar em soluções não triviais; não é sugerido o uso de critérios de tomada de decisão que não dependam da experiência do próprio projetista. Assim, entende-se que é necessário um aspecto adaptativo que compreenda, englobe e estimule as particularidades e peculiaridades do processo de desenvolvimento eletrônico quando na utilização em um modelo de referência.

2.3 DESENVOLVIMENTO DE SOFTWARE

Indubitavelmente, computadores são indispensáveis no mundo atual. Tais dispositivos são fisicamente compostos de placas eletrônicas (ou hardware) e não são, portanto, capazes de realizar tarefas sozinhos, ou seja, estas são realizadas por meio de uma série de dados expressos em programas, rotinas ou linguagens simbólicas, o que se pode conceituar como software (IEEE Standards Committee, 1990; Hassan; Markantonakis; Akram, 2016).

O software é responsável pelo processamento e transformação de informações, produzindo, gerenciando, adquirindo, modificando, exibindo ou as

transmitindo. Essas informações advêm, ou são geradas por sensores e/ou usuários em aplicações com diversos níveis de risco. Uma falha, portanto, pode acarretar desde pequenos inconvenientes pessoais até acidentes catastróficos. Dessa maneira, o desenvolvimento de software e seus domínios de aplicação devem ter um rigor, de modo a buscar a garantia da qualidade e confiabilidade (Pressman e Maxim, 2021).

2.3.1 Definições

A engenharia de software pode ser descrita como a aplicação de uma abordagem sistemática, disciplinada e quantificável para o desenvolvimento, operação e manutenção de um software, ou seja, a aplicação de engenharia ao software (IEEE Standards Committee, 1990; Alsaqqa; Sawalha; Abdel-Nabi, 2020).

Os softwares podem ser classificados em 3 tipos principais (IEEE Standards Committee, 1990; Teja, 2021):

- Software de sistema: conjunto de programas essenciais para coordenar os componentes eletrônicos de um computador. É projetado para facilitar a operação e manutenção de um sistema computacional e seus programas associados. Como exemplos, podem ser citados sistemas operacionais e compiladores (*e.g. Windows, Linux*);
- Software de aplicação: projetado para atender atividades e necessidades específicas dos usuários, ou seja, suas instruções são inseridas e processadas por meio dos softwares de sistema. Um exemplo simples é um editor de texto (*e.g. Microsoft Word*);
- Firmware⁵ ou software embarcado: referido de maneira vaga como uma camada intermediária entre o hardware e o software, ou seja, é um software de “baixo-nível”⁶ que controla diretamente o hardware. Ao transformar instruções de hardware em instruções de software, permite

⁵ Por muito tempo tal termo foi utilizado tanto para o hardware, quanto o software que o controla, devido à utilização de memórias do tipo ROM (não reprogramável). Advento que mudou com a utilização de EPROM (*Erasable Programmable ROM*) e memórias *Flash*, tornando o firmware uma parte dinâmica de um produto. (Hassan, Markantonakis, Akram, 2016; Teja, 2021).

⁶ Termo comumente usado para se referir a uma linguagem traduz diretamente instruções de programa em instruções de máquina (IEEE Standards Committee, 1990).

que o hardware performe tarefas específicas. Apesar do atrelamento a hardwares, seu desenvolvimento segue linguagens, ferramentas e metodologias utilizadas para desenvolver softwares.

Além das três categorias principais, Pressman e Maxim (2021) adicionam outras quatro categorias de software utilizadas atualmente:

- Software de engenharia/científico: programas para realização de “cálculo em massa” que abrangem astronomia, análise de estresse automotivo, dinâmica orbital, CADs, análise genética e meteorologia, entre outros;
- Software para linha de produtos: composto por componentes reutilizáveis e projetado para prover capacidades específicas de utilização por diversos clientes diferentes;
- Aplicações Web/aplicativos móveis: abrangem uma ampla variedade de aplicações, contemplando aplicativos voltados para navegadores, computação em nuvem, computação baseada em serviços e software residente em dispositivos móveis;
- Software de inteligência artificial: uso de heurísticas para solucionar problemas complexos que não são passíveis de computação ou de análise direta.

2.3.2 Processo de desenvolvimento

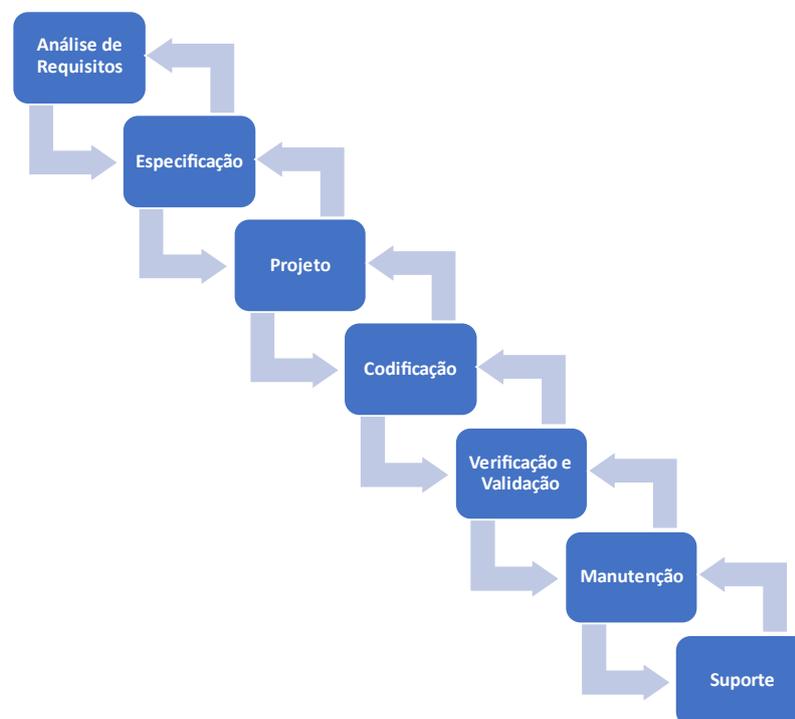
Um software é projetado para performar um conjunto particular de tarefas, com os graus mais altos de integridade, robustez e aceitação pelo usuário (Leau *et al.*, 2012). A fim de fornecer apoio a esses processos, muitos modelos de desenvolvimento de software foram criados ao longo dos anos. Segundo Saeed *et al.* (2019), os primeiros surgiram de maneira formal na década de 1960, com a utilização de fluxogramas em estágios. Cada modelo tem heurísticas e nomenclaturas próprias, assim como uma visão distinta do que é a qualidade do software. Não obstante, há características em comum para todos eles, apresentando:

1. Um mecanismo que transforma os requisitos em uma representação de projeto;
2. Uma representação dos componentes funcionais e interfaces do projeto;

3. Uma heurística para refinamento e particionamento;
4. Procedimento de avaliação de qualidade.

A ISO/IEC 12207–*Software Life Cycle Process* ou Processo do Ciclo de Vida do Software é um padrão internacional que define os processos e atividades associados ao ciclo de vida de software, da concepção ao descarte. Nele são descritas definições, terminologias e uma estrutura comuns com um ciclo de vida com fases principais, tais como (Figura 14): análise de requisitos, especificação, projeto, codificação, verificação e validação, instalação, manutenção e suporte.

Figura 14 - Processo do Ciclo de Vida do Software.



Fonte: Adaptado de Rai e Dhir (2014).

A referida norma, no entanto, não prevê orientação para implantação nem estabelece métodos, práticas ou ferramentas. Ela pode ser vista como uma orientação generalista utilizada pela indústria para projetar, desenvolver e testar softwares (Leau *et al.*, 2012; Akbar *et al.*, 2017; Aydan *et al.*, 2017).

Os modelos formais para desenvolvimento de software (apresentados adiante) indicam sequências e encadeamentos de atividades, práticas e ferramentas específicas. No entanto, independente do modelo, há alguns

passos generalistas para se desenvolver um software, tais como “Levantamento de Requisitos”, “Modelagem”, “Construção” e “Entrega”, os quais são apresentados a seguir.

2.3.2.1 *Levantamento de Requisitos*

O levantamento dos requisitos tem como objetivo compreender o que os *stakeholders* esperam do software a ser desenvolvido. É uma das principais tarefas de projeto, visto que um levantamento de requisitos mal feito pode levar a retrabalho, atrasos, artefatos de baixa qualidade, aumento do custo, clientes e usuários⁷ insatisfeitos e desmotivação dos desenvolvedores (Pressman e Maxim, 2021).

Para se estabelecer os requisitos, o processo é bastante semelhante àquele abordado no PDP, onde os *stakeholders* definem restrições de projeto predominantes e abordam as principais características e funções que levarão o projeto a cumprir seu objetivo. No entanto, há uma diferença na tangibilidade do software: por não haver comprometimento material, algumas definições específicas podem ocorrer ao longo do projeto.

Para Pressmann e Maxim (2021) alguns questionamentos sobre funcionalidade devem ser respondidos, tais como:

- Contexto: como o software a ser desenvolvido se ajusta a um sistema ou produto?
- Objetivos da informação: Quais saídas do software a ser desenvolvido são objetos de dados visíveis ao cliente?
- Utilização: Quais são as entradas necessárias de funcionamento?
- Funcionalidade: De que forma o software transforma os dados de entrada em dados de saída?
- Desempenho: Há algum desempenho ou características especiais que devem ser priorizados?

⁷ Cliente é quem requisita o software a ser construído, define os objetivos gerais os requisitos básicos e os recursos do negócio para o software. Usuário é o *stakeholder* que usa o software construído e define detalhes operacionais para atingir propósitos definidos (Pressman e Maxim, 2021).

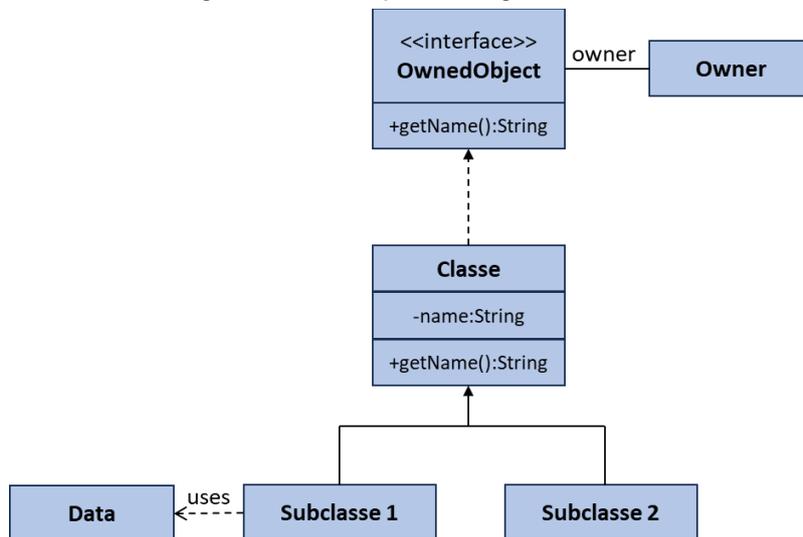
Uma prática comum para a representação de requisitos de software é por meio de uma modelagem de requisitos que combina as formas textual e diagramática, tornando mais fácil o entendimento e a comunicação, tal prática é comum em modelos ágeis (abordados adiante) em ferramentas como *story cards* e *use cases* (cartões de história e casos de uso).

2.3.2.2 Modelagem

A modelagem deve considerar a implementação de todos os requisitos explicitados, servindo de guia para posterior codificação e testes, e fornece uma visão completa de domínios de dados, funcional e comportamental do software.

Uma das formas mais utilizadas na geração de modelos é por meio da linguagem de modelagem unificada (UML), a qual serve para visualizar, especificar, construir e documentar um sistema de software. Essa linguagem é estabelecida como um padrão ISO para a indústria, com 13 distintos diagramas utilizados para representar classes, seus atributos, operações e relações na modelagem, como visto na Figura 15.

Figura 15 - Exemplo de diagrama UML.



Fonte: Adaptado de Pressman e Maxim (2021).

No decorrer do desenvolvimento de software, são necessárias diversas representações, ou seja, modelagens, que apresentam objetivos distintos, tais quais (Pressman e Maxim, 2021):

- Modelagem de requisitos: ilustra as necessidades do usuário;
- Modelagem de arquitetura: permite uma representação estrutural completa do software, seus subsistemas e componentes;
- Modelagem de interfaces: representa as interfaces internas e externas e do usuário;
- Modelagem de componentes: define cada um dos módulos que preenchem a arquitetura.

A modelagem de arquitetura compreende o início do movimento de uma visão macro do software até uma visão com detalhes necessários para implementar um sistema. Segundo Pressman e Maxim (2021), a “arquitetura de um sistema de software determina as suas qualidades e impacta o sistema durante todo o seu ciclo de vida”. Nesse momento, são definidas as relações entre os principais elementos estruturais do software, a organização de módulos, como é feita a interação entre os módulos e qual a estrutura de dados utilizada. As boas práticas na concepção da arquitetura preveem algumas características:

- Modularização do software, dividindo-o em componentes separadamente especificados e localizáveis;
- Independência dos módulos, reduzindo a probabilidade de que erros introduzidos na modificação de um módulo se propague a outros;
- Encapsulamento, no qual os módulos independentes trocam apenas informações necessárias para determinadas funções.

A modelagem de interfaces define como o software se comunica com o usuário, através da interface do usuário (UI – *user interface*), com outros sistemas, dispositivos ou redes, e entre os diversos componentes do próprio software. Dentre essas interfaces, a UI é de grande importância para validação com os clientes. É, portanto, uma boa prática avaliar a interface do usuário com o cliente antes de implementar o código. Alterar a UI nessa fase é menos custoso do que ter que retrabalhar posteriormente todas as entradas, saídas e código que a compõe (Pressman e Maxim, 2021).

Por fim, a última atividade abrange a modelagem de componentes, onde devem ser estabelecidos:

- As estruturas de dados para cada objeto de dado local;

- Os algoritmos para o processamento que deve ocorrer nos componentes;
- A interface que permite acesso a todas as operações dos componentes

Com a arquitetura, interfaces e componentes modelados, pode se considerar que o software está devidamente representado. Dessa maneira, pode ser iniciada a fase de construção.

2.3.2.3 *Construção*

A fase de construção engloba uma das atividades mais facilmente associada ao desenvolvimento de software, ou seja, a codificação, e em seguida são conduzidos testes. De acordo com Pressman e Maxim (2021), a codificação pode se dar de três maneiras:

1. Criação direta do código-fonte na linguagem de programação;
2. A geração automática de código-fonte usando uma representação intermediária semelhante a um projeto do componente a ser construído;
3. A geração automática de código executável usando uma linguagem de programação aplicada.

Enquanto o software é construído, devem ser escritos casos de testes para verificar funcionalidade e operacionalidade. Esses testes são realizados em distintos níveis (Pressman e Maxim, 2021):

- Testes de unidade: voltados para componentes;
- Testes de integração: realizados no sistema. Deve se garantir que inclusão de novas características não acarretará falhas que não eram antes observadas;
- Testes de validação: avaliação de atendimento de requisitos em incrementos ou sistema completo;
- Testes de aceitação: conduzidos pelo cliente, para avaliar os fatores e funções requisitados.

De acordo com Pressman e Maxim (2021), os testes podem demonstrar que “as funções do software estão funcionando de acordo com as especificações e que os requisitos relativos ao desempenho e ao comportamento parecem estar sendo atingidos”.

Normalmente após a codificação, o código passa ainda por um processo de refatoração, que consiste em reorganizar os componentes, simplificando o código sem alterar função ou comportamento do software. Esse processo é feito através da busca por redundâncias, elementos não utilizados, algoritmos desnecessários, estruturas de dados inadequadas e outras possíveis falhas de projeto (Pressman e Maxim, 2021).

Com o software devidamente testado e refatorado, é realizada a disponibilização ao cliente.

2.3.2.4 *Entrega*

A disponibilização, ou entrega, é o ato de disponibilizar o software ao cliente. Isso ocorre com frequências diferentes de acordo com o modelo de desenvolvimento empregado. No entanto, cada ciclo de entrega ainda conta com as ações de suporte, *feedback* e é acompanhado da documentação esperada.

Os modelos de software estão divididos entre duas filosofias principais: tradicional e ágil. Eles se distinguem desde a fase de requisitos, onde os modelos tradicionais têm um controle de documentação mais rígido, enquanto os modelos ágeis requerem menos formalismo, até a forma e frequência de entrega.

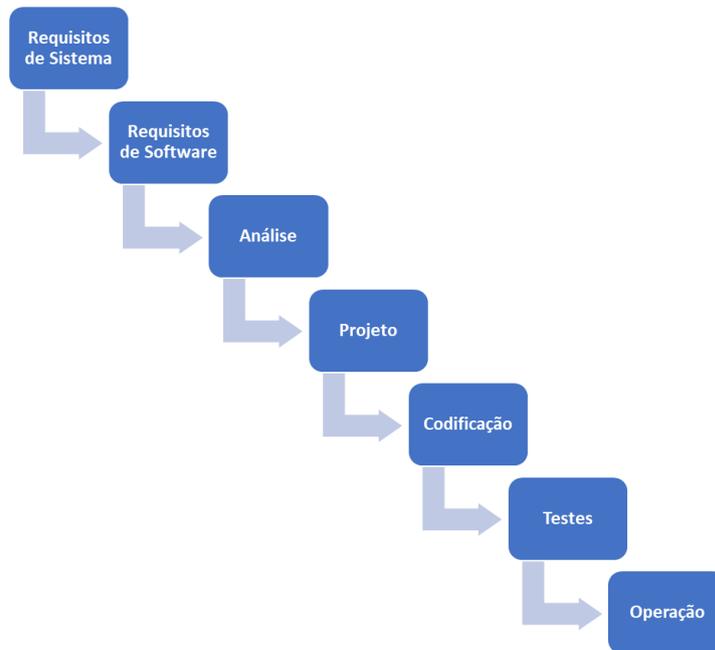
2.3.3 Modelos tradicionais

A maneira tradicional de desenvolver software envolve uma abordagem preditiva e prescritiva com planejamento prévio e extensivo, processos e fases predefinidos e ordenados, e definir e documentar um conjunto estável de requisitos logo ao início do projeto. Sua utilização é recomendada quando há uma fluidez e linearidade da comunicação da necessidade até a entrega do software, como, por exemplo em adaptações e aperfeiçoamentos de um sistema (Moniruzzaman e Hossain, 2013; Matharu *et al.*, 2015; Alsaqqa; Sawalha; Abdel-Nabi, 2020; Pressman e Maxim, 2021).

O modelo Cascata é o mais antigo, conhecido e difundido modelo tradicional de software. Caracteriza-se por uma execução sequencial e linear de estágios, de maneira que cada atividade anterior provê o insumo para a

próxima (Figura 16). É ideal para projetos nos quais os objetivos e requisitos são bastante claros desde o início e envolve uma extensa documentação (Akbar *et al.*, 2017; Banica *et al.*, 2017).

Figura 16 - Modelo Cascata.

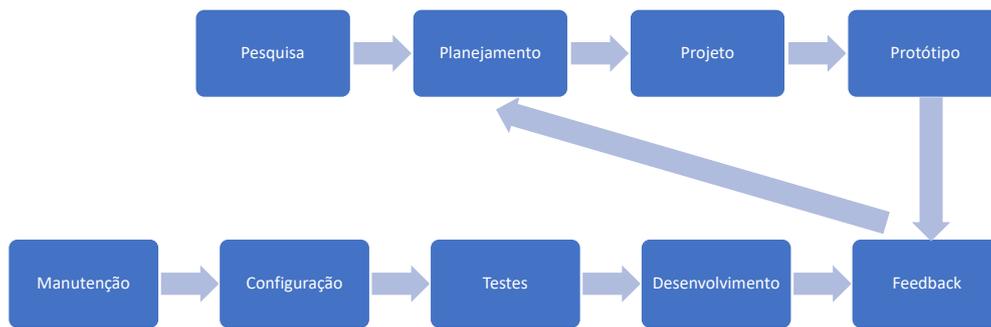


Fonte: adaptado de Saeed *et al.* (2019).

Ao se iniciar um projeto de software, é frequente que o cliente saiba algumas necessidades e objetivos do software, mas falte um entendimento mais específico para formar os requisitos.

No modelo da Prototipação (Figura 17), um protótipo é planejado a cada vez que se percebe uma função ambígua (principal ou derivada) para se observar o resultado e se obter definição mais ampla. De maneira geral, o desenvolvedor cria e testa funções específicas do programa em um protótipo com aspectos visíveis ao usuário para obter um *feedback* e, em seguida, volta ao planejamento principal com um refinamento de requisitos. Esse modelo é recomendável para projetos de porte grande ou quando há muitas incertezas, como em projetos inovadores (Rai e Dhir, 2014; Saeed *et al.*, 2019).

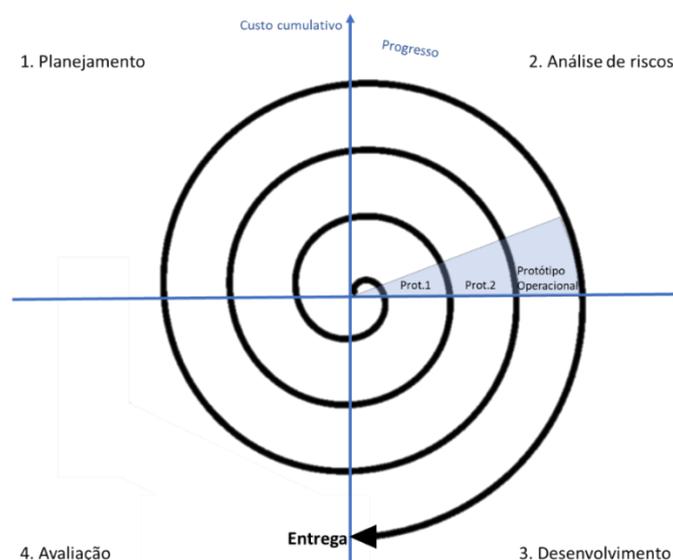
Figura 17 - Modelo da Prototipação.



Fonte: Adaptado de Saeed *et al.* (2019).

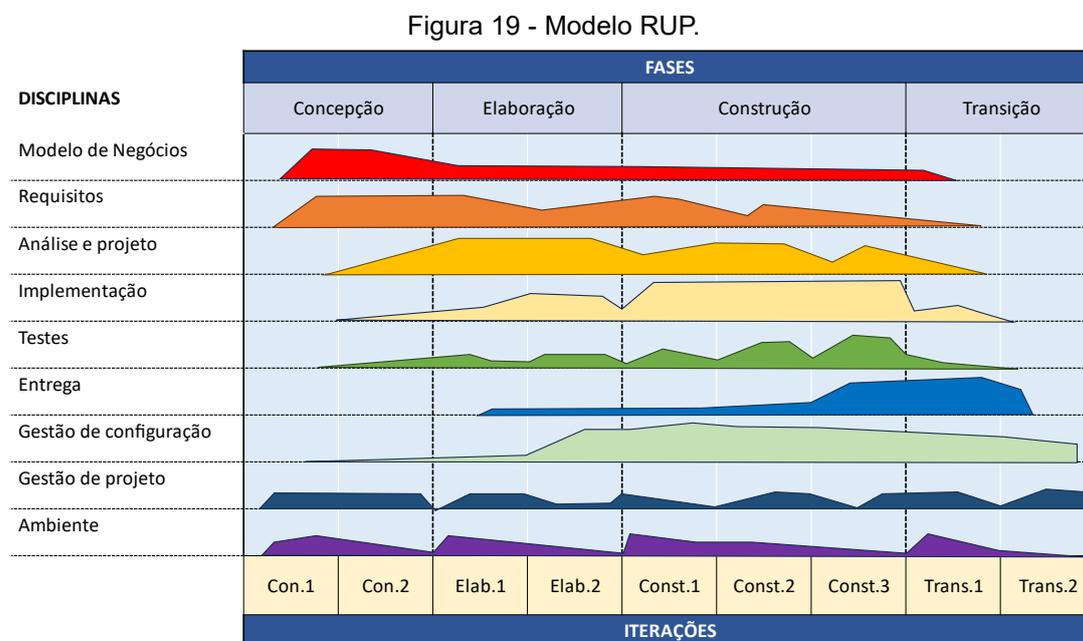
O modelo espiral parte do pressuposto que em projetos complexos há uma evolução de informações e do próprio projeto ao longo do tempo. Seguindo a espiral (Figura 18), o modelo funciona de uma maneira incremental de desenvolvimento e testes: iniciando no centro, o desenvolvimento segue as quatro etapas até que se estabeleça o primeiro protótipo, no qual se definem requisitos; o ciclo repete todos os estágios para evoluções mais sofisticadas até o protótipo final, que é normalmente a disponibilização do software. O modelo é indicado para projetos com características mutáveis e de possível crescimento, os quais se deve fazer avaliação constante de riscos (Akbar *et al.*, 2017; Saeed *et al.*, 2019; Pressman e Maxim, 2021).

Figura 18 - Modelo Espiral.



Fonte: adaptado de Saeed *et al.* (2019).

O processo unificado (RUP – do inglês *Rational Unified Process*) foi criado tentando combinar alguns recursos e características dos modelos tradicionais com alguns princípios ágeis. O modelo introduz um estilo de trabalho em paralelo, com quatro fases, conforme a Figura 19, no qual uma nova iteração se inicia antes do lançamento da iteração atual. Em cada fase o progresso é avaliado em diversas disciplinas de apoio para tomadas de decisões e produção de entregáveis que a completam. O RUP enfatiza o desenvolvimento e manutenção das modelagens⁸ como representações semânticas do sistema de software em desenvolvimento. É indicado para empresas de grande porte, pois o modelo exige uma equipe de desenvolvimento especializada (Anwar, 2014; Pressman e Maxim, 2021).



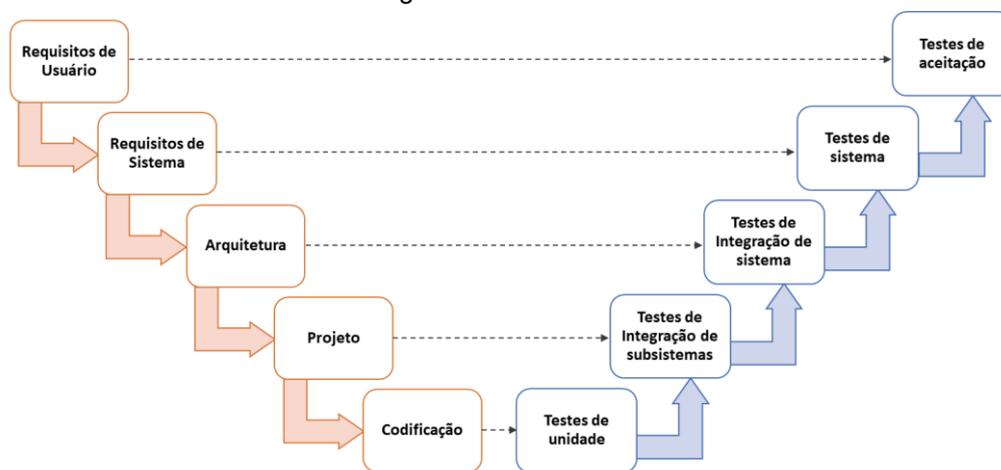
Fonte: adaptado de Anwar (2014).

O Modelo V foi proposto na década de 1980 como uma extensão do modelo cascata, ao combinar e associar cada fase de a uma fase de testes, com intuito de entender e lidar com complexidades de sistemas. Ao invés de se mover de maneira linear, os passos do processo se movem através de uma forma “V”. Como mostra a Figura 20, a porção à esquerda do “V” representa a fase de especificação e a direita a fase de testes. A frequência de testes é a

⁸ A modelagem é uma característica tão grande do processo que a UML, apresentada anteriormente, foi desenvolvida para apoiar as atividades do RUP (Pressman e Maxim, 2021).

característica mais marcante do modelo, o que auxilia a melhorar e eficiência e efetividade no desenvolvimento: o produto de cada fase deve ser checado e aprovado antes de avançar e uma vez que a codificação esteja encerrada, testes de unidade, integração e sistemas acontecem na sequência. A utilização do modelo V exige um número maior de recursos humanos, sendo mais adequado para projetos de portes médios e grandes e em grandes organizações (Balaji e Murugaiyan, 2012; Yadav, 2012; Akbar *et al.*, 2017; Saeed *et al.*, 2019).

Figura 20 - Modelo V.



Fonte: Adaptado de Saeed *et al.* (2019).

Como todas as áreas de tecnologia e negócios, a engenharia de software lida com constantes mudanças e evoluções. Um desafio para os desenvolvedores é a criação de uma lista de requisitos completa ao início do projeto, sendo esse um problema nos modelos tradicionais. Outra questão intrínseca nesses modelos é o baixo envolvimento dos *stakeholders* durante o desenvolvimento. Segundo Moniruzzaman e Hossain (2013) muitas vezes após 60% do tempo de desenvolvimento do projeto ainda não há software funcional para ser apresentado, testado e receber o *feedback* do cliente.

Os modelos apresentados são semelhantes a diversos modelos de PDP. O software, porém, se diferencia de outros produtos devido a sua intangibilidade. Essa característica permite desenvolvimentos mais rápidos, acomodação de mudança e *feedback* frequente do cliente, característica dos modelos ágeis apresentados a seguir.

2.3.4 Modelos ágeis

Agilidade é a habilidade de entender e responder ao mercado e se manter inventivo em um ambiente instável. No ponto de vista do desenvolvimento de software, ágil não é um modelo em si, mas uma abordagem com uma série de valores e princípios para promover respostas rápidas a mudanças de ambiente, restrições ou requisitos (Moniruzzaman e Hossain, 2013; Alsaqqa; Sawalha; Abdel-Nabi, 2020).

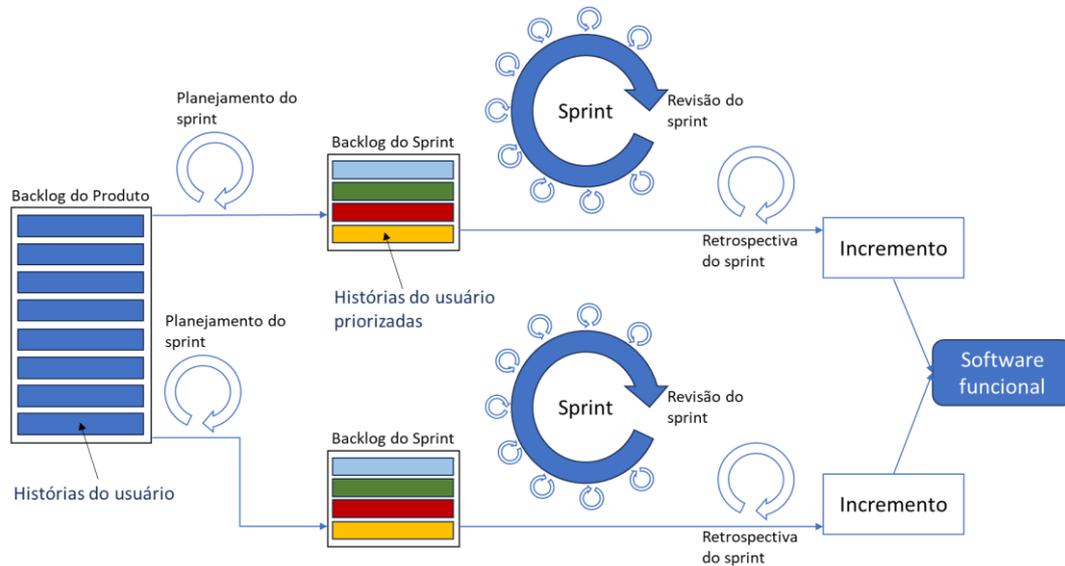
A abordagem ágil se originou em 2001 com uma aliança de 17 consultores de engenharia de software, os quais publicaram um manifesto composto de 4 valores e 12 princípios: o manifesto de desenvolvimento de software ágil. O Anexos B e C apresentam, respectivamente, os valores e princípios do manifesto ágil (Rai e Dhir, 2014; Alsaqqa; Sawalha; Abdel-Nabi, 2020).

De maneira geral, os modelos baseados na abordagem ágil se baseiam na ideia de se dividir um projeto em incrementos e de forma iterativa, com documentação mínima, ciclos rápidos, centrado nas equipes e com envolvimento frequente com o cliente através de entregas e *feedbacks* regulares e contínuos. Tal contato frequente mitiga um dos maiores problemas em projetos: o que o cliente aceitará ao final do projeto se difere do que foi demandado no início. (Leau *et al.*, 2012; Moniruzzaman e Hossain, 2013; Rai e Dhir, 2014; Pressman e Maxim, 2021).

Dentre seus modelos, o Scrum é um modelo de processo iterativo e incremental para produzir ou gerenciar projetos, o qual, apesar de ter sido criado no início dos anos 1990 (antes da publicação do manifesto ágil), foi adotado como modelo ágil de desenvolvimento de software. Suas práticas de desenvolvimento são baseadas em reuniões diárias, nas quais se decide quais atividades retiradas do *backlog* do produto (lista priorizada de requisitos que agregam valor para o cliente) serão desenvolvidas nos próximos *sprints* (iterações curtas) para se tornarem incrementos de software (Figura 21). Uma vez que todos os itens do *backlog* se tornaram incrementos e os requisitos e objetivos do projeto cumpridos atendam ao requerido pelo cliente, é realizada a entrega do software. A versatilidade e facilidade de uso do Scrum o torna

amplamente aplicável em quase todos os domínios (Matharu *et al.*, 2015; Akbar *et al.*, 2017; Alsaqqa; Sawalha; Abdel-Nabi, 2020; Pressman e Maxim, 2021)

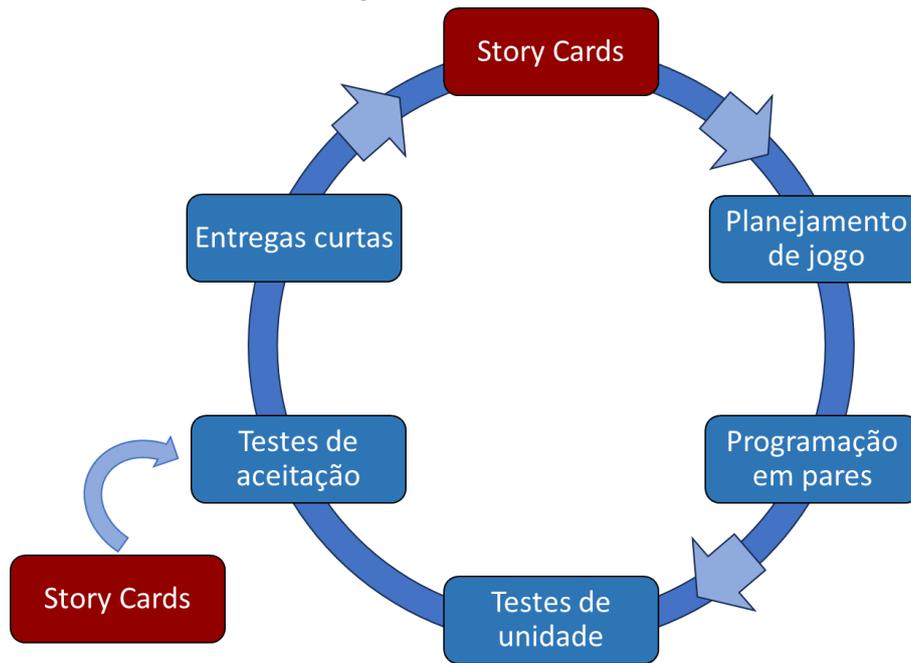
Figura 21 - Modelo Scrum.



Fonte: adaptado de Alsaqqa, Sawalha e Abdel-nabi (2020).

O *Extreme Programming* (XP) foi criado para resolver questões-chave entendidas como limitações em modelos tradicionais: o não atingimento dos requisitos dos clientes; modelos criados com foco em softwares de grande porte; o cancelamento do projeto antes do mesmo ir para produção; possíveis desentendimentos de negócio, como o software não resolver os problemas para o qual foi desenvolvido. As principais características do XP (Figura 22) são: requisitos representados como *Story Cards*; iniciação do desenvolvimento do software com um projeto simples, com posterior adição de funcionalidades (planejamento do jogo); interação com o cliente em ciclos de *feedback* pré-estabelecidos; desenvolvimento direcionado por testes; testes de unidades automatizados; testes de aceitação obtidos de *Story Cards*; Refatoração; e programação em pares. A agilidade do modelo decorre do processo de fragmentar o desenvolvimento em pequenas partes, gerando pequenas tarefas facilmente executáveis pelos programadores (Al-Tarawneh *et al.*, 2011; Matharu *et al.*, 2015; Akbar *et al.*, 2017; Pressman e Maxim, 2021).

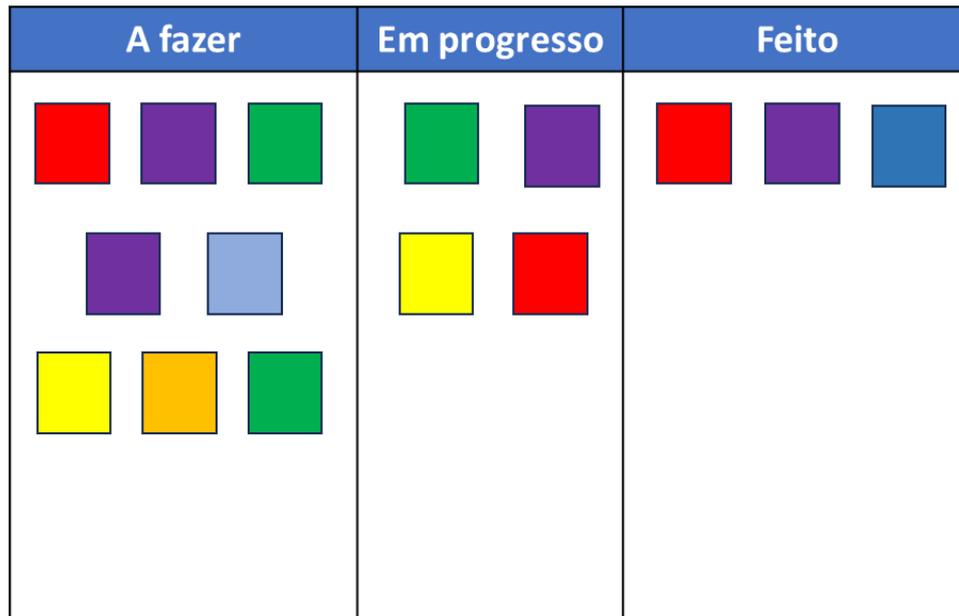
Figura 22 - Modelo XP.



Fonte: adaptado de Matharu *et al.* (2015).

O modelo Kanban foi implementado inicialmente na indústria automotiva Toyota como um conceito relacionado ao *Lean* e produção *just-in-time*. De prática de engenharia industrial, foi adaptado ao desenvolvimento de software e é utilizado dentro da abordagem ágil. Pode ser visto como um sistema para visualizar trabalho, controlar e melhorar o fluxo de execução de atividades, reduzir o desperdício e maximizar o valor entregue ao cliente. A principal característica do modelo é o quadro Kanban, ilustrado na Figura 23, para visualização do fluxo em: trabalho a fazer, trabalho em progresso e trabalho feito. Sua utilização auxilia na entrega contínua de incrementos de software, redução de desperdício, limitação do trabalho em progresso (de acordo com a capacidade da equipe) e a busca da melhoria contínua. Apesar do crescente interesse e utilização, não há definições padronizadas no modelo para como desenvolver um sistema de software: como especificar um atributo, quais atividades representam melhor o processo, como representar tarefas ou como lidar com exceções e riscos. Dessa maneira, pode-se combinar o Kanban a outras práticas de desenvolvimento ágil para adicionar mais disciplina a elas (Corona e Pani, 2013; Matharu *et al.*, 2015; Pressman e Maxim, 2021).

Figura 23 - Quadro Kanban.

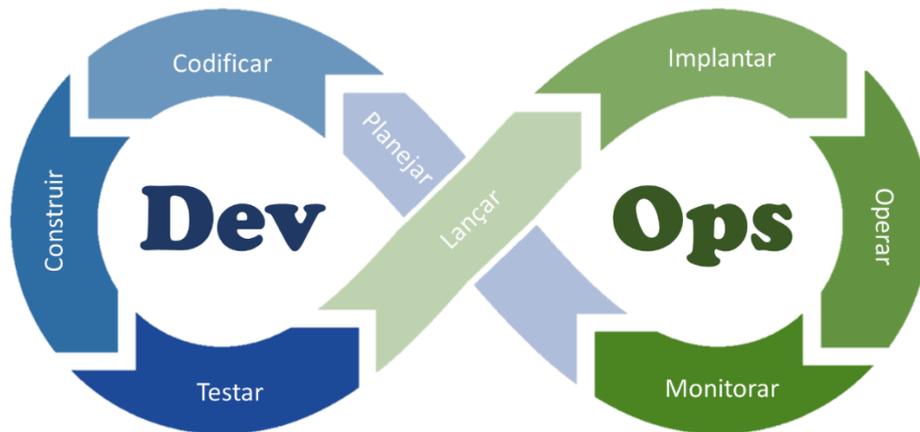


Fonte: adaptado de Matharu *et al.* (2015).

O DevOps (junção das palavras “*development*” e “*operations*”) é um modelo que busca integrar todas as funções do software em ciclos contínuos, do desenvolvimento às operações. O modelo aplica princípios advindos tanto da abordagem ágil, quanto do *Lean* e é considerada uma extensão dos modelos ágeis⁹, buscando validação e entrega rápida de softwares. Os ciclos contínuos, tais como mostrados na Figura 24, podem ser divididas em: fases de desenvolvimento (planejar, codificar, construir, testar) e fases de operações (lançar, implantar, operar e monitorar). A utilização do DevOps busca melhorar a experiência dos clientes, acelerando reação às mudanças advindas de necessidades. Ao proporcionar maior transparência baseada em comunicação aberta entre as equipes de desenvolvimento e operação, reduz-se o tempo de entrega do software para produção (Banica *et al.*, 2017; Gokarna e Singh, 2021; Pressman e Maxim, 2021).

⁹ Embora o conceito DevOps tenha sido introduzido em 2009, há, segundo Banica *et al.* (2017) diferentes opiniões em relação à sua definição – há pesquisas que a chamam de *framework*, ferramenta ou até mesmo como uma terceira geração de metodologia de desenvolvimento de software.

Figura 24 - Ciclo DEVOPS.



Fonte: adaptado de Gokarna e Singh (2021).

2.3.5 Considerações sobre o Desenvolvimento de Software

Os modelos descritos anteriormente são alguns dos principais modelos ágeis encontrados atualmente para o desenvolvimento de software. Podem ser recomendados quando se busca: simplificar ou desburocratizar o desenvolvimento do software; se aproximar do cliente; reduzir a quantidade de documentação; e garantir entregas contínuas com menor tempo de desenvolvimento (Leau *et al.*, 2012; Alsaqqa; Sawalha; Abdel-Nabi, 2020; Pressman e Maxim, 2021).

No entanto, não há um modelo que sirva para todos os casos. Segundo Pressman e Maxim (2021), “mesmo hoje em dia, a maioria das metodologias de projeto de software carece da profundidade, flexibilidade e natureza quantitativa que normalmente estão associadas às disciplinas mais clássicas de engenharia de projeto”. Mesmo a abordagem ágil comporta abreviar soluções, a disciplina continua sendo essencial para que se entenda o que deve ser entregue e seu respectivo valor ao cliente.

Em grandes projetos, com grandes investimentos, e requisitos estáveis, muitas vezes a utilização dos modelos tradicionais ainda pode ser recomendável. No entanto, a resposta à mudança, como em projetos criativos, pode ser vista como um critério com grande peso para se escolher um modelo ágil.

Independente do porte, todo software está atrelado a uma camada física: um computador pessoal, um microcontrolador ou mesmo um periférico. Ao se desenvolver um produto que busca entregar tanto o software, quanto

partes tangíveis, seu processo de desenvolvimento deve ser integrado a outras áreas do conhecimento. A seção a seguir mostra como a engenharia de sistema auxilia na integração entre distintas áreas do conhecimento

2.4 ENGENHARIA DE SISTEMAS

Atualmente, dadas poucas exceções, os produtos utilizados no cotidiano são frutos da união de conhecimentos advindos de diversas áreas do conhecimento. Um computador, por exemplo resulta, de maneira simplificada, em desenvolvimentos de engenheiros eletrônicos, de software e mecânicos – respectivamente suas placas, seus programas e seu gabinete. Tais produtos não podem ser separados em partes independentes sem perder suas características essenciais: uma placa não funciona sem software, um gabinete depende dos componentes os quais irá acondicionar e o software necessita um meio físico para ser processado, exibido e permitir interação. Quando o valor principal está na relação entre as partes, se caracteriza como um sistema. Para o Conselho Internacional em Engenharia de Sistemas (INCOSE, 2004) um sistema é um conjunto de diferentes elementos que juntos produzem resultados, os quais não são obtidos pelos elementos de forma separada (NASA, 2007; Cloutier, Baldwin e Bone, 2015).

Quando os sistemas crescem em complexidade e aumentam a demanda de interações, as disciplinas de engenharia especializadas não proporcionam uma visão geral adequada, pois estão normalmente focadas em seus próprios desenvolvimentos. Ao se retornar ao exemplo do produto (ou sistema) computador, enquanto as engenharias eletrônica, de software e mecânica contribuem para o desenvolvimento das partes, uma outra área de conhecimento pode ser utilizada para comunicar uma visão compartilhada do sistema desenvolvido, buscando evitar omissões, confusões e riscos associados à integração dessas disciplinas: a engenharia de sistemas (ES) (INCOSE, 2004).

A ES foi introduzida por volta dos anos 1950, impulsionada pela corrida espacial e guerra fria, como um braço das engenharias em setores militares e civis, especialmente para desenvolvimento de missões espaciais pela NASA. Uma das primeiras citações do termo ocorreu em 1957 no jornal “*Electrical Engineering*” como um conceito lógico e compreensivo para adaptar os

produtos de grande complexidade aos requisitos e limitações de seus usuários. Décadas mais tarde, em 1998, alguns órgãos normativos, tais como a Associação Americana de Padrões Nacionais (ANSI), a Aliança de Indústria Eletrônicas (EIA) e a Associação Internacional de Engenharia Eletrônica e Elétrica (IEEE) juntaram esforços para publicar um padrão de processos para engenharia de sistemas (INCOSE, 2004; Cloutier, Baldwin e Bone, 2015).

2.4.1 Processos técnicos

Uma característica da ES é o pensamento sistemático e reducionista: ao se dividir um assunto complexo em porções menores, essas podem ser tratadas de maneira metódica e progressiva. Utiliza-se uma abordagem decomposição–recomposição: produtos¹⁰ são decompostos¹¹ em níveis consecutivos na hierarquia desde o sistema até componentes de projeto. Esses componentes são, então, recompostos em subsistemas e finalmente no sistema completo. Os efeitos da composição são validados e verificados a cada nível.

Com uma visão abrangente, a ES conduz um processo iterativo que compreende: (1) entender um problema antes de tentar solucioná-lo; (2) examinar alternativas de solução; e (3) verificar se a solução escolhida está adequada antes de prosseguir com as próximas atividades (Cloutier, Baldwin e Bone, 2015).

Nota-se a semelhança do processo de Engenharia de Sistemas e de Desenvolvimento de Produtos. Segundo Kannengiesser e Gero (2022), isso se deve ao fato de que as abordagens da ES foram desenvolvidas originalmente por engenheiros nas áreas militar e aeronáutica com formações, em sua maioria, em engenharia mecânica e elétrica. Corroborando com as semelhanças, uma forma de organizar o processo desenvolvimento é através da decomposição do desenvolvimento em atividades básicas, também chamados de processos técnicos. De uma maneira geral pode-se determinar

10 Para a SE, um produto é resultado de um desdobramento que realiza funções relacionadas ao sistema. De acordo com a maturidade do sistema pode ser desde um relatório técnico, até a parte física que forma um subsistema (NASA, 2007).

11 Esse processo é conhecido como PBS (*Products Breakdown Structure* – Estrutura de desdobramento de produtos): desdobramento hierárquico de produtos - itens de hardware, software e informação pertencentes ao sistema (NASA, 2007).

oito processos técnicos¹² para desenvolver um sistema (INCOSE, 2015; Kannengiesser e Gero, 2022):

- Definição de Expectativas e Requisitos de *Stakeholders*;
- Definição de Requisitos de Sistema;
- Definição de Arquitetura;
- Definição de Projetação;
- Implementação;
- Integração;
- Verificação;
- Validação.

Ressalta-se que, devido à abordagem decomposição-recomposição, muitos dos processos visam o desenvolvimento de produtos, os quais são posteriormente recompostos para o desenvolvimento de um sistema. Os tópicos a seguir apresentam de maneira resumida o conteúdo dos processos listados da ES.

2.4.1.1 *Definição de Expectativas e Requisitos de Stakeholders*

O primeiro processo técnico diz respeito a entender e documentar de maneira clara os objetivos da missão, ou seja, elicitando as necessidades, desejos e premissas dos *stakeholders*. A saída são os requisitos dos *stakeholders*, que irão governar o desenvolvimento do sistema (NASA, 2007).

Uma ferramenta utilizada a partir dessa fase é o Conceito de Operações¹³ (ConOps): descrição, de maneira visual, do comportamento e características do sistema de um ponto de vista operacional, ou seja, quais as ações e saídas do sistema são geradas por determinados estímulos externos. O ConOps permite a construção de cenários para obter o primeiro esclarecimento sobre: o uso do sistema, o ambiente de operação e o impacto do sistema no ambiente, e as interfaces do sistema ou produtos. Esses cenários auxiliam no entendimento dos requisitos dos *stakeholders*.

¹² O INCOSE propõe 14 processos técnicos, os quais englobam desde análise de negócios até o descarte. Na presente dissertação optou-se por listar os processos mais condizentes ao desenvolvimento propriamente dito.

¹³ O ConOps é um elemento que evolui como ciclo de desenvolvimento. Pode descrever as expectativas dos *Stakeholders*, Requisitos ou a Arquitetura do projeto (NASA, 2007).

Os requisitos dos *stakeholders* são uma visão de capacidades desejadas por um perfil de usuário que deverão ainda, no próximo processo técnico, incorporar uma visão técnica, ou seja, serem transformados em requisitos do sistema (INCOSE, 2015; NASA, 2007).

2.4.1.2 *Definição de Requisitos do Sistema*

Através de discussões aprofundadas e análises, a engenharia de sistemas busca converter as expectativas do *stakeholders* em uma definição do problema na forma de uma lista de requisitos. Tais requisitos são orientados com uma visão técnica e irão especificar as características, atributos, funções e desempenho que o sistema deve ter para cumprir as expectativas definidas no processo anterior (INCOSE, 2015).

Utilizando a abordagem decomposição-recomposição, os requisitos são levantados maneira iterativa do nível de sistema até os componentes mais básicos e passíveis de implementação, formando um conjunto contendo elementos de hardware¹⁴, software e interfaces do sistema (NASA, 2007; Cloutier, Baldwin e Bone, 2015),

O conjunto de requisitos final deve ser escrito de uma maneira que sejam: claros, únicos, consistentes, individuais, verificáveis, rastreáveis, não redundantes e não vinculados a uma implementação particular. Esses requisitos são a *baseline*¹⁵ inicial do sistema e usados como fundação para alocação de funções e, posteriormente, para definir a arquitetura, as soluções de projeto e também os produtos habilitadores¹⁶ necessários. A utilização de uma equipe multidisciplinar para auxiliar na análise é um fator-chave para a descoberta de problemas (INCOSE, 2004; NASA, 2007; Cloutier, Baldwin e Bone, 2015).

¹⁴ O termo hardware aqui é atribuído a qualquer parte física do sistema.

¹⁵ *Baselines* são documentações que descrevem diretrizes do projeto, tais como requisitos, funções, arquitetura. Suas versões são estáveis e mudanças e evoluções devem ser aprovadas e monitoradas de maneira formal (NASA, 2007).

¹⁶ Produtos que facilitam o processo e utilização do sistema ou produto final ao longo de qualquer fase do seu ciclo de vida, mas não são parte integrante do mesmo.

2.4.1.3 Definição de Arquitetura

A arquitetura do sistema é uma visão abstrata e global de elementos de sistema com foco em cumprir o ConOps. Define, de uma maneira diagramática, a decomposição de elementos ou recursos do sistema até o ponto necessário para implementar uma solução de projeto, ou seja, cumprir diretamente suas funções e também definir suas interfaces internas e externas. Pode ser desenvolvida através de uma análise funcional que (INCOSE, 2004; NASA, 2007; Cloutier, Baldwin e Bone, 2015):

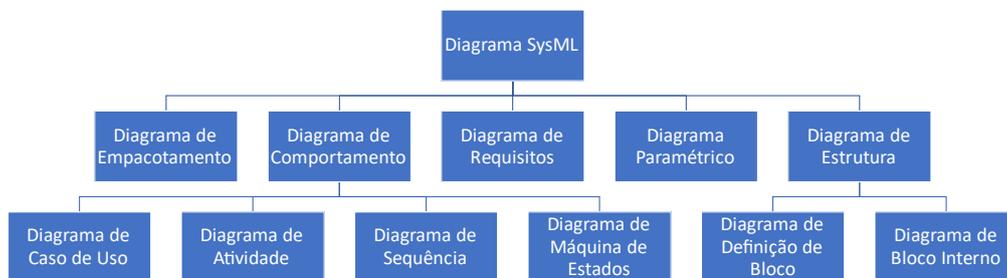
- Converta os requisitos do sistema em funções que devem ser executadas para cumpri-los;
- Decomponha e aloque as funções de hardware e software até o nível mais baixo da hierarquia do produto;
- Identifique e descreva as interfaces funcionais e de subsistemas;
- Represente as funções e interfaces através de um modelo ou diagrama de blocos.

Uma ferramenta utilizada para representar a arquitetura de sistemas é a linguagem SysML (Linguagem de Modelagem de Sistemas). Considerada uma extensão da UML, a SysML busca trazer uma linguagem abstrata comum na construção de sistemas, incluindo componentes para hardware, software, dados, procedimentos e instalações, através dos diagramas (Figura 25) (HAUSE, 2006; Mcharek *et al.*, 2019):

- Diagrama de Empacotamento: utilizado para organizar o modelo;
- Diagrama de Comportamento:
 - Diagrama de Caso de Uso: descreve descrição em alto nível da funcionalidade do sistema;
 - Diagrama de Atividade: descreve fluxo de data e controle entre atividades;
 - Diagrama de Sequência: representa a interação entre partes colaborativas de um sistema;
 - Diagrama de Máquina de Estados: descreve as transições e ações do estado de um sistema (ou suas partes) em resposta a eventos;

- Diagrama de Requisitos: descreve relações, hierarquia e rastreabilidade e de requisitos;
- Diagrama Paramétrico: descreve atributos e restrições variáveis em propriedades do sistema;
- Diagrama de Estrutura:
 - Diagrama de Definição de Bloco: descreve a hierarquia do sistema e classificação de sistemas e componentes;
 - Diagrama de Bloco Interno: descreve a estrutura interna do sistema em suas partes, portas e conexões;

Figura 25 - Diagramas do SysML.



Fonte: adaptado de INCOSE (2015).

A definição de arquitetura é um processo criativo, recursivo e iterativo que combina o entendimento dos objetivos e restrições de projeto com conhecimento em meios técnicos para estabelecer os produtos. Uma boa representação da arquitetura pode revelar problemas ou prover informações que não foram levantadas inicialmente (NASA, 2007; Cloutier, Baldwin e Bone, 2015).

2.4.1.4 Definição de Projetação

De maneira geral, a Definição de Projetação é a última etapa para a transformação das expectativas dos *stakeholders* em uma solução de projeto. Nesse processo técnico devem-se estabelecer as soluções físicas, de hardware e software no nível de detalhe suficiente para implementação do produto (Cloutier, Baldwin e Bone, 2015). Os passos básicos para a atividade são (NASA, 2007):

1. Proposição de alternativas de soluções para a arquitetura do sistema;

2. Realização de estudos de *trade-off* para análise das alternativas propostas;
3. Escolha de uma alternativa mais bem avaliada;
4. Detalhamento da alternativa até uma solução final de projeção;
5. Geração de especificações para posterior produção e verificação do produto.

O primeiro passo é a proposição de alternativas de solução para a arquitetura do sistema. Esse passo é conduzido pelo engenheiro de sistemas com apoio constante dos engenheiros de disciplinas. O objetivo é incluir todos as disciplinas envolvidas para identificar soluções a todos os elementos condutores de projeto (INCOSE, 2004).

Através de análises de *trade-off*, as alternativas são selecionadas levando-se em consideração requisitos, expectativas e tecnologia disponível. Essa análise permite avaliar e escolher a melhor alternativa, que deve, então, ser definida com maiores detalhes nos elementos do sistema (hardware, software, informação), interfaces internas e externas e produtos habilitadores. O processo continua até a projeção completa da solução para satisfazer os requisitos técnicos. Essa solução estabelecerá o *baseline* de projeto, o qual permite realizar a alternativa escolhida (INCOSE, 2004; NASA, 2007).

As atividades de Projeção requerem um trabalho próximo do engenheiro de sistemas com a equipe – disciplinas de engenharia. Com o produto e seu *baseline* especificados, os próximos processos técnicos se iniciam através da implementação dos produtos do sistema.

2.4.1.5 *Implementação*

A implementação (ou realização) é o processo que transforma os planos e conceitos em produtos propriamente ditos. A depender do projeto ou fase do ciclo de vida, o produto pode ser desde relatórios, modelos e simulações até produtos tangíveis (INCOSE, 2004; NASA, 2007).

Enquanto os produtos do sistema são desenvolvidos (desde os requisitos até o detalhamento na projeção), deve-se considerar sempre como os mesmos serão realizados e quais os critérios para sua aprovação. Esse processo pode ocorrer de três distintas formas (NASA, 2007):

- **Aquisição:** produtos adquiridos devem garantir os requisitos de projeto e passar por critérios de verificação e avaliação. Normalmente há normas e padrões aplicáveis para apoiar essas atividades;
- **Fabricação/Codificação:** o produto deve ser produzido de acordo com os requisitos especificados, documentações geradas e normas e padrões aplicáveis. Deve-se garantir a prontidão de produtos habilitadores, como softwares necessários, disponibilidade de máquinas e operadores treinados;
- **Reutilização:** produtos desenvolvidos anteriormente para outros projetos de sistemas podem ser utilizados no sistema em desenvolvimento. Devem, no entanto, ser minuciosamente analisados para garantir a aplicabilidade no projeto atual e passar pelos mesmos processos de validação e verificação que produtos adquiridos ou fabricados.

Independentemente da forma selecionada, toda a documentação pertinente deve ser gerada, incluindo desenhos, documentação de projeto, listas de código, descrição de modelos, procedimentos, manuais de operação e manutenção. Uma vez os produtos implementados, pode-se seguir com a etapa integração.

2.4.1.6 *Integração*

O propósito do processo técnico de Integração é assegurar que a combinação dos elementos atinge o resultado requerido. Para prover tal combinação, as interfaces internas e externas devem estar implementadas. As interfaces podem ter elementos lógicos, físicos, elétricos, mecânicos e humanos e devem ser definidas e controladas desde os primeiros processos do desenvolvimento para prevenir problemas na integração.

Utilizando a referida abordagem decomposição – recomposição, produtos que foram decompostos nas camadas inferiores do PBS são integrados para formar produtos das camadas superiores (subsistemas), e assegurar que o funcionamento integrado é adequado. De forma iterativa, isso ocorre até que, ao final desse processo técnico, se forme o sistema completo, o qual irá cumprir os objetivos estabelecidos (NASA, 2007; Cloutier, Baldwin e Bone, 2015).

2.4.1.7 Verificação

A Verificação determina se o sistema é construído conforme a especificação. De maneira simples, visa responder à pergunta: “construímos o sistema de maneira certa?” (Cloutier, Baldwin e Bone, 2015).

Esse processo técnico tem relação direta com os requisitos de sistema e pode ser executado em diversos estágios de desenvolvimento. Com os produtos realizados, há quatro tipos de verificação possíveis: (NASA, 2007; Cloutier, Baldwin e Bone, 2015):

- Análise: utilização de modelos matemáticos, algoritmos, simulações e técnicas analíticas para prever a conformidade do projeto em relação aos requisitos;
- Demonstração: operação real de um item do sistema para evidenciar que as funções foram realizadas dentro de cenários específicos, podendo ser realizada por modelos físicos ou *mockups*. Se diferencia dos testes por não haver aquisição de dados detalhados;
- Inspeção: exame visual, auditivo, olfativo, tátil, paliativo ou manipulativo de um produto para verificar características de projeto ou identificação. O item não precisa estar em operação para que essa verificação ocorra;
- Teste: utilização de um produto para obter dados e verificar desempenho no cumprimento de suas funções. Pode ser conduzido nos produtos finais, *protoboards* ou outros protótipos.

Um plano deve especificar qual o tipo de verificação será utilizado, as condições e ambiente, assim como o critério de aprovação. As primeiras verificações podem ser feitas pelo próprio desenvolvedor, enquanto verificações finais costumam ter participação do cliente. A última verificação, chamada de verificação ponta-a-ponta, é a mais significativa em qualquer projeto: costuma ser um teste realizado com a configuração e condições mais realistas possíveis e atesta que o sistema, não só os produtos ou componentes, está conforme os requisitos, especificações e documentos descritivos (NASA, 2007; Cloutier, Baldwin e Bone, 2015).

2.4.1.8 Validação

A Validação, diferentemente da verificação, não mede o atendimento aos requisitos, mas sim às expectativas dos *stakeholders*. Busca responder à questão “nós construímos o sistema certo?”. Mesmo que um sistema passe todas as verificações e cumpra cada requisito, o mesmo pode ser reprovado na validação.

Como esse processo é relacionado a uma das primeiras etapas de projeto de sistema, as expectativas dos *stakeholders*, o mesmo é atrelado a um significativo grau de incerteza. Dessa maneira, é necessário que haja validações prévias desde o ConOps. Muitas vezes os *stakeholders* expressam o que eles querem, não o que precisam; ou ainda têm problema de comunicar o que querem (Cloutier, Baldwin e Bone, 2015).

Os testes são conduzidos sob condições realistas (ou simuladas) dos produtos de acordo com sua fase no ciclo de vida. No caso do sistema ou produtos finais, a mesma só pode ser executada pelos *stakeholders* com o produto finalizado (*i.e.* construído). Muitas vezes um projeto de sistema é descartado por falhar em cumprir as necessidades originais dos *stakeholders*. Se isso é feito no processo técnico de validação, o custo para mudança pode inviabilizar a continuidade do projeto.

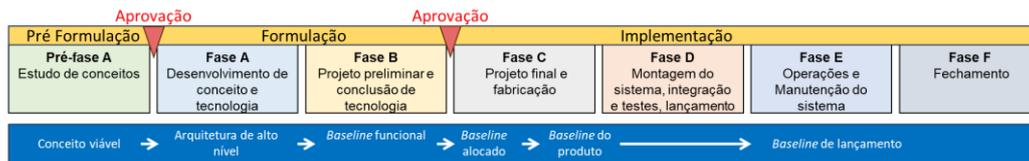
Assim como outras áreas do conhecimento, há diversos modelos propostos para se executar a engenharia de sistema. As subseções a seguir apresentam alguns desses modelos.

2.4.2 Modelo da NASA

Para executar a engenharia de sistemas em um projeto, a NASA (2007) propõe um processo de ES (Figura 26) constituído por:

- Pré-fase A: Estudo de conceitos;
- Fase A: Desenvolvimento de conceito e tecnologia;
- Fase B: Projeto preliminar e conclusão de tecnologia;
- Fase C: Projeto final e fabricação;
- Fase D: Montagem do sistema, integração e testes, lançamento;
- Fase E: Operações e Manutenção do sistema;
- Fase F: Fechamento.

Figura 26 - Processo de ES da NASA.



Fonte: adaptado de NASA (2007)

A Pré-fase A tem como propósito produzir uma gama de ideias e alternativas para missões nas quais novos projetos podem ser selecionados.

Essas ideias são amadurecidas na Fase A por meio da determinação dos requisitos funcionais e de desempenho, juntamente com arquiteturas e conceitos, conforme estudos iterativos de *trade-off*.

A seguir, durante a Fase B, se define um *baseline* funcional de projeto: através de uma abordagem *top-down*, se avaliam os requisitos desde o nível de sistema até os mais básicos, em nível de componente, e se propõe um conjunto completo de solução capaz de cumprir os requisitos de cada nível.

Durante a Fase C, o objetivo é estabelecer o *baseline* de produto completo: informações para fabricar ou produzir hardware e codificar o software em preparação para a integração do sistema e seus subsistemas associados (incluindo produtos habilitadores necessários para operação).

As atividades da fase D são executadas para montar, integrar, testar e lançar o sistema, que deve se mostrar capaz de cumprir a proposta para a qual foi criado.

As últimas fases do processo contêm atividades posteriores ao desenvolvimento: na fase E, as atividades são executadas para a operação do sistema, fornecendo o suporte necessário ao cliente; já, o propósito da fase F é implementar o descomissionamento e descarte.

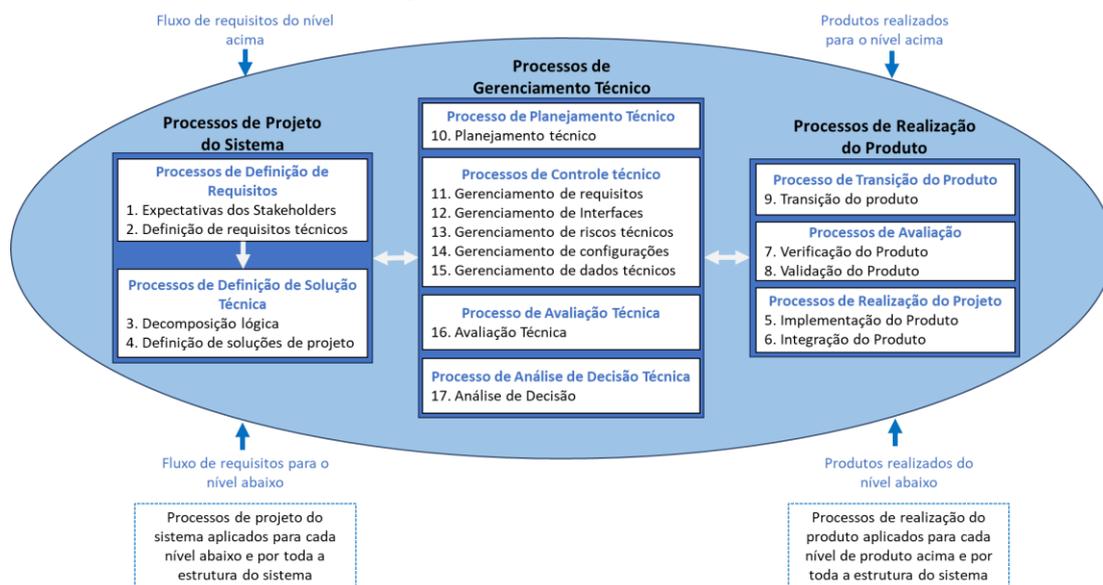
Uma característica presente no processo sugerido pela NASA é a ocorrência periódica de revisões de aprovação para seguimento do projeto. O Anexo D resume o conteúdo dessas revisões.

Aliado a isso, para guiar o desenvolvimento de um sistema, a NASA (2020) sugere um modelo contendo um conjunto de processos técnicos, ao qual é dado o nome de “Locomotiva da ES” (Figura 27). A Locomotiva da ES é utilizada de forma recursiva através de um PBS, onde as primeiras camadas

hierárquicas do produto representam o sistema e subsistemas, e as camadas de desdobramento mais básicas representam componentes; cada elemento de determinada camada passa a ser considerado um produto próprio. Dessa maneira, três agrupamentos de processos técnicos são aplicados a cada camada na estrutura do sistema:

- Processos de Projetação de sistema: desenvolvimento “*top-down*” de cada produto na estrutura do sistema;
- Processos de realização de produto: realização “*bottom-up*” de cada produto na estrutura do sistema;
- Processos de gerenciamento técnico: planejamento, avaliação e controle dos processos acima e auxílio para decisões técnicas.

Figura 27 - Locomotiva da ES.



Fonte: adaptado de NASA (2007).

2.4.3 Modelo do INCOSE

O Conselho Internacional em Engenharia de Sistemas, ou INCOSE (2004), propõe um processo baseado em 4 fases, que acompanham o desenvolvimento evolucionário do sistema. Tais fases, ilustradas na Figura 28, são divididas em:

- Fase 0: Exploração de Conceitos;
- Fase I: Definição do programa e redução de riscos;
- Fase II: Desenvolvimento de Engenharia e Manufatura;

- Fase III: Produção, Lançamento e Suporte de Operações.

Figura 28 - Fases e tarefas-chave no processo do INCOSE.

	Fase 0	Fase I	Fase II	Fase III	
Necessidades	Exploração de conceitos	Definição do programa e redução de riscos	Desenvolvimento de Engenharia e Manufatura	Produção, Lançamento e Suporte de Operações	Descarte
	1. Análise de Sistema 2. Definição de Requisitos 3. Projeto Conceitual 4. Avaliação de tecnologia e riscos 5. Custos, cronograma e desempenho preliminares do conceito	6. Atualização do Projeto Conceitual 7. Trade-offs de subsistemas 8. Projeto Preliminar 9. Prototipação, testes e avaliação 10. Integração de considerações de manufatura e manutenção	11. Projeto Detalhado 12. Desenvolvimento 13. Gerenciamento de riscos 14. Testes de desenvolvimento e avaliação 15. Integração de Sistema, Testes e Avaliação 16. Verificação de processo de Manufatura	17. Verificação de taxa de produção 18. Testes de operação e avaliação 19. Lançamento 20. Suporte de operação e atualização 21. Planejamento de reposição	
← Definição Evolucionária de Requisitos →					
Tarefas-chave	<ul style="list-style-type: none"> Objetivos Missão Requisitos funcionais Conceitos e arquiteturas concorrentes Alocação de requisitos Trade-offs e síntese Definição de conceito Estimativa de custo e cronograma 	<ul style="list-style-type: none"> Revisão de conceito Modelagem e simulação de alta fidelidade Projetos de subsistemas integrados Redigir especificações de alto nível Plano de desenvolvimento Análise de custo e risco Planejamento de mitigação de risco 	<ul style="list-style-type: none"> Integração de programa técnico Medição de desempenho técnico Revisões de projeto Realocação de requisitos Documentação do sistema Controle de interface Controle de mudanças Participação de Desenvolvimento Integrado de Produto (IPDT) 	<ul style="list-style-type: none"> Controle de mudanças por manufatura Análise de testes Verificação de projeto Engenharia de suporte e solução de problemas Manutenção Treinamento para suporte Plano de desenvolvimento evolucionário 	

Fonte: adaptado de INCOSE (2004).

Após um estudo de viabilidade o projeto do sistema se inicia na Fase 0, onde esforço é dedicado para definir todos os aspectos do sistema: análise do sistema, requisitos, arquitetura, modelagem e análise dos subsistemas; proposição de conceitos, estudos de *trade-off* no sistema e subsistemas; e análise de conceito de subsistemas, visando esclarecer seus potenciais e limitações.

A Fase I ocorre como parte de uma estratégia de gerenciamento de risco para comprovar se o sistema proposto é funcional antes do comprometimento de recursos para continuar o desenvolvimento e a posterior implementação: as atividades buscam evoluir o sistema e alocar esforço em tarefas de projeção para desenvolver um produto tangível ou utilizável.

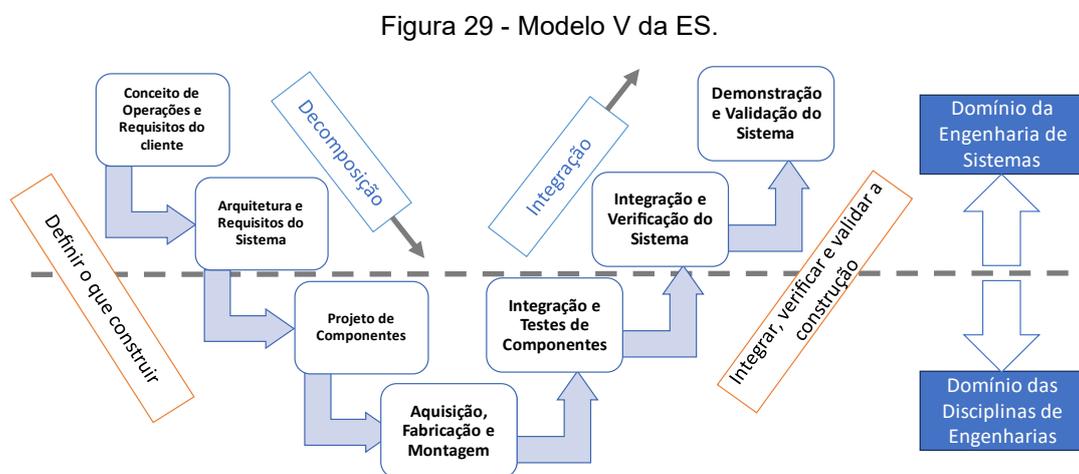
A Fase II é a que exercita mais os processos de engenharia de sistema, com atividades de testes em verificação, integração e validação em nível de componentes e no sistema integrado. São fabricados modelos de engenharia e verificação de requisitos a fim de garantir que o sistema como um todo esteja funcionando conforme conceituado.

Por fim, na fase III, com o sistema já desenvolvido, o foco direcionado à solução de problemas que podem ocorrer na manufatura, montagem, integração e verificação no sistema já lançado. Ainda é feito o plano de projeto para descarte/descomissionamento.

Utilizando uma abordagem próxima de um modelo V, o processo de engenharia de sistemas é utilizado iterativamente a cada fase para gerar descrições mais detalhadas do sistema: se inicia com uma representação das primeiras camadas do sistema e decompõe essas representações até as camadas mais básicas para a solução de problemas.

2.4.4 Modelo V de Desenvolvimento

De maneira similar ao apresentado anteriormente para desenvolvimento de software, o modelo V para a ES é proposto pelos autores Cloutier, Baldwin e Bone (2015), com 7 fases focadas no desenvolvimento propriamente dito. Este modelo, ilustrado na Figura 29, segundo os autores, representa o processo básico para desenvolver um sistema.



Fonte: adaptado de Cloutier, Baldwin e Bone (2015).

No lado esquerdo do “V” se situam atividades voltadas à projeção e construção. Se inicia com a definição de um ConOPs, cujos cenários auxiliam a estabelecer os requisitos funcionais e de desempenho dos clientes. Tais cenários são convertidos em requisitos do sistema, o que permite que se defina uma arquitetura que mostre como as principais partes do sistema são unidas entre si e como o sistema interage com o ambiente operacional.

No centro do “V” se situam as atividades de Aquisição, Fabricação e montagem – esse centro se une ao lado esquerdo, pois enquanto as atividades de projeção e construção ocorrem, devem ser consideradas noções de como o sistema será adquirido ou fabricado e montado.

O lado direito do “V” representa a integração e recomposição. Após a implementação, o sistema como um todo é integrado por meio da união dos componentes físicos aos funcionais e lógicos. É, então, realizada uma verificação para os requisitos nos componentes de hardware e software. Os testes de validação seguem para confirmar que o sistema completo em seu ambiente satisfaz as necessidades dos clientes. Finalmente, após a validação, a fase de produção se inicia com a responsabilidade do sistema sendo entregue ao setor de manufatura.

2.4.5 Considerações da Engenharia de Sistemas

Os modelos e processos técnicos apresentados visam ao desenvolvimento de sistemas conforme um rigor analítico. Quando se entende o que o sistema deve cumprir, é possível conceber diversas maneiras de se atingir o resultado esperado. Todo o processo de engenharia de sistemas busca mitigar a chance do sistema – e dos produtos que o compõem – falhar ao longo do desenvolvimento incremental (NASA, 2007).

Identifica-se que, apesar de ter sido criada para projetos de grande porte, diversos conceitos e processos apresentados por modelos da ES podem ser adaptados no desenvolvimento de produtos de maior ou menor complexidade, auxiliando no desenvolvimento de sistemas (ou produtos multidisciplinares) de qualidade. Segundo o INCOSE (2004), a customização da ES deve envolver um processo que identifique os processos padrões e proveja requisitos para estabelecimento, manutenção e melhoria dos processos. Especialmente em projetos ou programas de menor porte, as atividades da ES podem ser conduzidas de uma maneira mais informal.

2.5 CONSIDERAÇÕES DO CAPÍTULO

Conforme a revisão de literatura apresentada, algumas considerações podem ser traçadas a respeito dos processos de desenvolvimento:

- O PDP, mais aplicável para a indústria metal mecânica, é um processo bastante linear e, devido a um custo crescente de mudanças, tem um grande esforço criativo para embasar as decisões tomadas nas primeiras etapas;
- O desenvolvimento eletrônico é um processo majoritariamente linear. Conforme identificado, as primeiras atividades sugeridas se destinam à escolha de componentes. Esforços criativos para integrar os componentes em uma PCI concisa são realizados somente na segunda metade do processo;
- O desenvolvimento de software pode ser realizado de maneira linear, nos modelos tradicionais, ou utilizando os modelos ágeis com a divisão em diversos ciclos de desenvolvimento. Os modelos ágeis buscam a rápida resposta à mudança, possível devido à intangibilidade do software;
- A Engenharia de Sistemas é uma abordagem generalista e integrativa, na qual se busca produzir um todo que não é dominado dentro de só uma disciplina. Sua execução, conforme os modelos apresentados, demanda uma equipe especializada e alto grau de formalismo.

Ao se considerar que os constituintes de qualquer sistema são justamente os resultados do desenvolvimento de diferentes disciplinas técnicas, a engenharia de sistemas pode ser utilizada para preencher as lacunas e as interfaces entre elas. No entanto, apesar de buscar a visão compartilhada do sistema em desenvolvimento, e evitar problemas de integração, a ES não guia ou conduz o desenvolvimento de um hardware ou um código.

Kannengiesser e Gero (2022) investigaram como a ES diverge de outros processos de desenvolvimento. Os autores afirmam que, apesar da sucessão da SE em relação a abordagens de PDP e de software, há muitas etapas que não se sincronizam entre as abordagens. Os autores afirmam que, ao entender como o processo da ES se diferencia de outros métodos de projeto, pode guiar futuros desenvolvimentos.

Com viés semelhante, McCarthy (2021) executou um mapeamento de atividades de modelos de desenvolvimento aplicados a hardware e software e

os comparou com atividades que ocorrem no processo do INCOSE. Seus estudos apontam que, apesar de haver uma semelhança na natureza das atividades, elas seguem processos distintos. No entanto, o processo proposto pela engenharia de sistemas pode, segundo a autora, servir como de base de customização para processos desenvolvimento em qualquer domínio.

Ainda, a recente revolução digital proporcionou maior acesso a novas tecnologias e o subsequente desenvolvimento de produtos multidisciplinares em organizações de pequeno e médio porte. Ao se escolher como base de customização um processo difundido e integrativo como a ES com um menor formalismo (conforme proposto pelo INCOSE, 2004) podem-se englobar atividades, terminologias e características condizentes de modelos distintos de desenvolvimento em um modelo apropriado para o desenvolvimento de produtos multidisciplinares. O próximo capítulo trata da proposição de tal modelo.

3 PESQUISA DE CAMPO E PROPOSIÇÃO DA METODOLOGIA

Segundo Pahl *et al.* (2007), a principal tarefa de engenheiros é aplicar os conhecimentos técnicos e científicos para a solução de problemas. Os capítulos anteriores apresentaram o problema de pesquisa e fundamentaram conhecimentos técnicos e científicos a respeito da projeção de soluções, respectivamente.

O presente capítulo aborda a utilização e integração desse conhecimento a fim de contribuir com o desenvolvimento de produtos multidisciplinares por meio da proposição de um modelo de desenvolvimento e seu processo. As seções que seguem explicitam a delimitação do escopo para, então, ser apresentado o modelo propriamente dito.

3.1 RESTRIÇÃO DE ESCOPO

A busca por um modelo abrangente para o desenvolvimento de qualquer tipo de solução dificilmente trará uma aplicação satisfatória para todos os casos – normalmente empresas têm seus próprios modelos e processos de desenvolvimento. Nessa linha, o modelo proposto não se direciona a uma empresa em particular, mas busca atender a demandas de desenvolvimento que apresentem as seguintes características:

- Produtos do tipo criativo: novidades para a empresa e mercado (vide item 2.1.2);
- Produtos dos ramos ciber-físico, *smart* ou mecatrônico: produtos com características tangíveis e intangíveis (vide item 2.1.2);
- Desenvolvimento que demande diferentes áreas de conhecimento¹⁷: no presente caso a engenharia mecânica, engenharia eletrônica e engenharia de software;
- Empresas fornecedoras de tecnologia: instituições de pesquisa ou centros de pesquisa e desenvolvimento¹⁸;

¹⁷ Apesar do modelo ser proposto para essas três áreas, pode ser posteriormente expandido para agregar outras áreas do conhecimento.

¹⁸ Mais informações para tipos de empresas e cadeia de tecnologia em Rozenfeld *et al.* (2007).

- Atividades relacionadas ao desenvolvimento de soluções propriamente dito. Não engloba atividades relacionadas ao planejamento ou produção comercial¹⁹.

Dentre os estudos relacionados ao processo de desenvolvimento de produtos e projetos na engenharia, identifica-se que termos iguais podem apresentar distintos significados ou aplicações de acordo com autores, idioma original (e variação na tradução), épocas ou área de conhecimento. Por exemplo, para Back *et al.* (2008) produtos são utensílios produzidos industrialmente, já para a NASA (Hirshorn, Voss e Bromley, 2017) produtos são as saídas das fases, podendo assumir diversas formas. A fim de evitar ambiguidade, os seguintes termos foram adotados para a construção do modelo proposto:

- Disciplinas de engenharia (DE): áreas de conhecimento utilizadas para desenvolvimento de produtos;
- Produtos: resultado de desenvolvimento de uma área de conhecimento ou DE, como desenvolvimento mecânico, eletrônico ou software. Por exemplo: um pistão pneumático; uma placa de circuito impresso, ou uma aplicação de software, respectivamente;
- Desenvolvimento mecânico: desenvolvimento de produtos tangíveis sem componentes eletrônicos de acionamento ou controle. Tem por característica ser um processo majoritariamente criativo com pontuais escolhas de componentes comerciais;
- Desenvolvimento eletrônico: desenvolvimento de produtos físicos com componentes eletrônicos a fim de prover comunicação, leitura de dados, acionamentos e controle. Tem por característica ser um processo regido por escolha de componentes comerciais e pontuais decisões criativas;
- Desenvolvimento de software: desenvolvimento de produtos intangíveis para leitura, visualização e manipulação de dados. Tem por característica ser um processo criativo e incremental, em ciclos;
- Engenharia de Sistemas (ES): área de conhecimento da engenharia cujo objetivo é o desenvolvimento de sistemas;

¹⁹ Tais atividades são, de acordo com Reinert (2017), mais relacionadas às estratégias da organização, fabricação e lançamento do produto.

- Produto Multidisciplinar: tem como resultado um sistema, pois decorre da integração de produtos advindos de DEs, visando cumprir objetivos estabelecidos por *stakeholders*. Possui características tangíveis e intangíveis, e é considerado a entrega final da equipe que aplicará o modelo proposto.

3.2 PROPOSTA DO MODELO DE DESENVOLVIMENTO DE PRODUTO MULTIDISCIPLINAR

A construção do Modelo de Desenvolvimento de Produto Multidisciplinar (MDPM) parte de algumas recomendações da literatura acerca da construção de modelos de referência:

- Rozenfeld *et al.* (2007) sugerem, a partir da definição de um ciclo de vida, a descrição de atividades, recursos, informações, fases e responsabilidades;
- Romano (2003) propõe a estrutura de um modelo de referência com a descrição das entradas, saídas, controles e mecanismos de cada tarefa (estrutura conhecida nos diagramas IDEF0) tendo em vista o desenvolvimento de produtos multidisciplinares;
- Ao se considerar o produto multidisciplinar como um sistema, o grupo INCOSE (2004) sugere partir de um processo base, incluir os produtos específicos, revisões a serem conduzidas, grau de formalismo esperado no processo, número de iterações esperadas e critério de sucesso para cada atividade.

Com base em tais recomendações, o MDPM propõe unir as boas práticas da ES e DEs, utilizando um processo de ES para nortear o desenvolvimento de produto multidisciplinar, bem como modelos advindos das práticas de projeção das DEs para auxiliar no desenvolvimento dos produtos específicos, com entradas, saídas e mecanismos.

Uma fonte adicional utilizada para fortalecer a problemática de pesquisa e esclarecer necessidades para potenciais usuários é através de uma pesquisa exploratória. Tal pesquisa, apresentada no apêndice A, traz algumas recomendações apontadas para um modelo multidisciplinar, tais como: definição de pontos de integração, processos e entregáveis específicos e pessoas de referência por área de desenvolvimento.

3.2.1 Ciclo de desenvolvimento do MDPM

O ciclo de desenvolvimento do MDPM apresenta divisão em fases, as quais são inspirados nos processos técnicos apresentados no item 2.4.1, adaptados do INCOSE (2015):

- Definição de Expectativas: elicitación e documentação das necessidades, desejos, capacidades e premissas dos *stakeholders*;
- Definição de Requisitos: conversão das expectativas dos *stakeholders* em um conjunto de requisitos claros, únicos, consistentes, individuais, verificáveis, rastreáveis, não redundantes e não vinculados a uma implementação particular;
- Definição de Arquitetura: decomposição e modelagem de elementos ou recursos dos produtos, de maneira gráfica, até o ponto necessário para implementar uma solução de projeto;
- Projetação: estabelecimento, por meio de análises de *trade-off*, de conceitos de soluções físicas, de hardware e software dos produtos no nível de detalhe suficiente para implementação;
- Realização: evolução de planos e conceitos a produtos tangíveis e/ou utilizáveis;
- Integração: junção dos produtos, testes de interfaces e funcionamento conjunto apropriado;
- Verificação: determinação do cumprimento de requisitos através de análise, demonstração, inspeção ou testes;
- Validação: testes para averiguar se o produto multidisciplinar construído cumpre os objetivos dos Stakeholders.

Com a formalização das fases, são identificados os domínios das mesmas, conforme o Quadro 1:

- DE: fases em que o principal foco é o desenvolvimento dos produtos são domínio das disciplinas de engenharia;
- ES: fases nas quais o principal foco é a integração dos produtos são domínio da engenharia de sistemas.

Quadro 1 - Domínios do MDPM.

Fase	Domínio
Definição de Expectativas	ES/DE
Definição de Requisitos	DE
Definição de Arquitetura	DE
Projetação	DE
Realização	DE
Integração	ES
Verificação	ES
Validação	ES

Fonte: o autor.

Assume-se que a fase de “Definição de Expectativas” é de domínio tanto da Engenharia de Sistemas, quanto das Disciplinas de Engenharia, pois, ao tratar o produto multidisciplinar como um todo, as demandas, necessidades e expectativas dos *stakeholders* devem ser esclarecidas para todos os envolvidos. Ainda, as fases de Integração, Verificação e Validação²⁰ são formalizadas e explícitas na ES, ou seja, são fases que são executadas uma vez que os produtos das disciplinas de engenharia estão finalizados (salvo retrabalhos) de maneira individual e a integração em um sistema único passa a ser o principal foco. No desenvolvimento das fases com o domínio das DEs, utilizam-se modelos recomendados pela literatura.

Para o desenvolvimento mecânico, foi selecionado o modelo PRODIP proposto por Back *et al.* (2008), apresentado em 2.1.3. De acordo com o escopo, a macrofase de “Elaboração de projeto do produto” abrange as seguintes fases:

- Projeto Informacional: estabelecimento das especificações de projeto;
- Projeto Conceitual: geração de conceito de produto;
- Projeto Preliminar: definição da configuração do produto;
- Projeto Detalhado: documentação detalhada do produto e processo de manufatura.

Para o desenvolvimento eletrônico, como mencionado, não foi encontrado um modelo de referência na literatura para todo o ciclo,

²⁰ Alguns modelos de PDP, como o proposto por Romano (2023) e Back *et al.* (2008) contam com fases com a nomenclatura de validação ou outras obras. O modelo de Pahl *et al.* (2007) traz recomendações para integração. No entanto, essas fases serão estabelecidas de acordo com as diretrizes da engenharia de sistemas.

especialmente para elicitación e priorização de necessidades e/ou requisitos. A base utilizada partiu dos passos propostos por Russ (2022) e Rigo (2019), discutidos em 2.2.2, denominados aqui como Processo de Desenvolvimento de Eletrônica (PDE). Nesse estudo, as fases de projeção podem ser divididas em:

- Seleção de componentes;
- Esquemático;
- *Layout* da placa;
- Arquivos de fabricação.

Por fim, para o desenvolvimento de software utilizaram-se práticas advindas da abordagem ágil, em especial do modelo XP²¹. Como as fases do modelo proposto apresentam uma representação de linear, a divisão ocorreu de acordo com as atividades apresentadas em 2.3.3:

- Requisitos de projeto;
- Modelagem;
- Construção;
- Entrega.

O Quadro 2 apresenta a relação das fases de ES e dos modelos adotados das DEs.

Quadro 2 - Relação das fases entre os modelos.

Fase	Desenvolvimento Mecânico	Desenvolvimento Eletrônico	Desenvolvimento de Software
Definição de Expectativas	Projeto Informacional	-	Requisitos de projeto
Definição de Requisitos	Projeto Informacional	-	Requisitos de projeto
Definição de Arquitetura	Projeto conceitual	Seleção de componentes	Modelagem
Projeção	Projeto conceitual/Projeto preliminar	Esquemático Definição de <i>Layout</i>	Construção
Realização	Projeto preliminar/Projeto detalhado	Arquivos de fabricação	Entrega

Fonte: o autor.

²¹ Como prática já adotada em modelos ágeis, ferramentas e atividades advindas de outros modelos podem ser incorporadas.

Como pode ser visto no Quadro 2, para o PDE não há correspondentes às duas primeiras fases, pois não foram encontradas na literatura recomendações para elucidar as expectativas ou selecionar os requisitos do usuário. Dessa maneira, os mecanismos para essas fases serão replicados do PDP.

3.2.2 Apresentação do MDPM

A definição de modelos de desenvolvimento e fases advindas das DEs, assim como a determinação dos domínios, permite que se visualize um processo cujos fluxos mostrem a relação entre as DEs e a ES. A Figura 30 representa de forma resumida o MDPM.

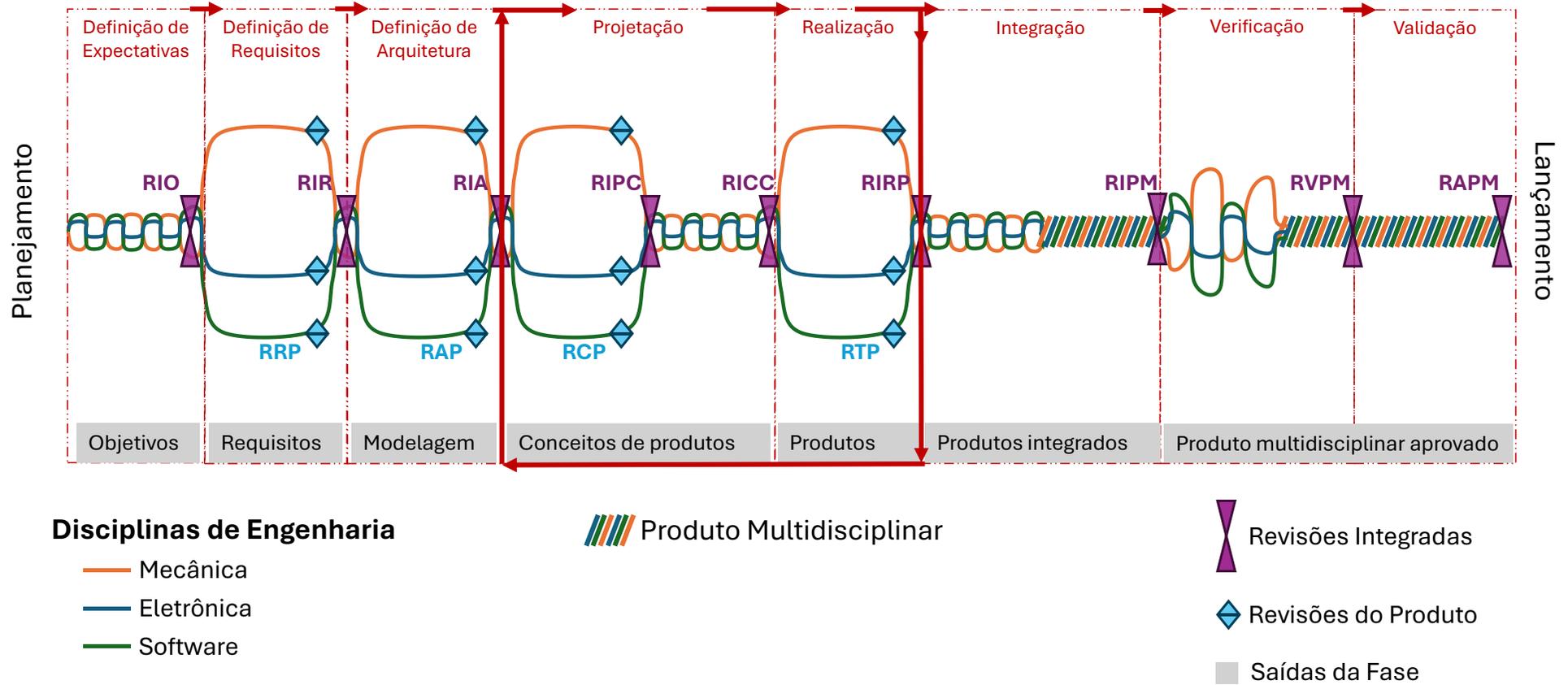
A execução do modelo requer uma equipe composta por:

- Equipes de DE: responsáveis por executar os mecanismos de desenvolvimento de cada DE;
- Líder de DE: representante da equipe da DE o qual, além de conduzir o desenvolvimento do produto, interage diretamente com o engenheiro de sistemas;
- Engenheiro de sistemas: responsável por executar os mecanismos da ES, integrar os produtos e conduzir a um produto multidisciplinar que supra as necessidades dos *stakeholders*. Interage diretamente com os líderes das DEs e com o cliente.

O modelo busca tratar dos processos tanto dos produtos de DEs, quanto da integração desses em um produto multidisciplinar. Nota-se, na Figura 30, a ocorrência de revisões que tratam dos produtos e revisões que os analisam de forma integrada, visando sua compatibilização e recursividade (essa relação será explicitada posteriormente por meio de uma representação conforme o IDEF0).

Figura 30 - Representação do MDPM.

Modelo de Desenvolvimento de Produto Multidisciplinar



Fonte: o autor.

3.2.2.1 Revisões do produto

De acordo com a nomenclatura adotada, os produtos são os resultados de desenvolvimento das DEs envolvidas. Por conseguinte, as revisões propostas objetivam verificar se as saídas (detalhadas nas próximas seções), ou entregáveis, de cada DE, permitem, por fase de desenvolvimento, integração aos demais produtos, mitigando eventual retrabalho. Tendo a abordagem ágil como base, essas revisões são propostas em um formato que evite sobrecarga de documentação, mas mantém o conteúdo necessário para aferir e formalizar a evolução dos produtos. O MDPM propõe 4 revisões de produtos, de acordo com o Quadro 3.

Quadro 3 - Revisões de Produto do MDPM.

RRP – Revisão de Requisitos do Produto
Conteúdo: Lista contendo descrição, razão e método de verificação para os requisitos do produto.
Ocorrência: Ao final da fase de Definição de Requisitos, anterior à Revisão Integrada de Requisitos.
Critério de sucesso: Requisitos claros, concisos e verificáveis.
Saída: Requisitos do produto aprovados para integração com demais DEs.
RAP – Revisão de Arquitetura do Produto
Formato: Modelagem diagramática de funções e elementos do produto.
Ocorrência: Ao final da fase de Definição de Arquitetura, antes da Revisão Integrada de Arquitetura.
Critério de sucesso: Demonstração de funções e elementos do produto, incluindo interfaces internas e externas com outros produtos.
Saída: Modelagem diagramática do produto e suas interfaces aprovada para integração com demais DEs.
RCP – Revisão de Conceito do Produto
Conteúdo: Registro visual de conceito ou protótipos iniciais.
Ocorrência: Durante a fase de Projetação, antes da Revisão de Integrada de Conceitos.
Critério de sucesso: Os conceitos devem se mostrar viáveis para seguir o desenvolvimento.
Saída: Conceito inicial do produto aprovado para integração com conceitos das demais DEs.
RTP – Revisão de Testes do Produto
Conteúdo: Registro visual e relatórios de testes.
Ocorrência: Ao final da fase de Realização, antes da Revisão de Integração de Produtos
Critério de sucesso: Testes dos produtos devem se mostrar bem-sucedidos, os requisitos dos produtos devem se mostrar minimamente atendidos.
Saída: Produto aprovado para integração com demais produtos de DEs.

Fonte: o autor.

As revisões de cada produto devem ocorrer em reuniões conduzidas pelo líder da DE, com participação da equipe da DE e o resultado dessa revisão deve ser aprovado pelo engenheiro de sistemas. A continuidade do desenvolvimento do produto de cada DE se dá pela aprovação da revisão. Os *templates* propostos para a documentação do resultado das revisões dos produtos da DEs estão disponíveis no Apêndice B.

3.2.2.2 *Revisões integradas*

As revisões integradas buscam assegurar que os produtos desenvolvidos em cada fase do ciclo se integrem da maneira esperada em um sistema e, posteriormente, um produto multidisciplinar. A proposição de tais revisões é inspirada nas revisões de sistema sugeridas pelo *handbook* de Engenharia de Sistemas da NASA (2017), apresentadas no Anexo D. O Quadro 4 apresenta as características dessas revisões.

Quadro 4 - Revisões integradas do MDPM.

RIO – Revisão Integrada de Objetivos
Conteúdo: Lista contendo as necessidades e objetivos dos <i>stakeholders</i> e proposta inicial de conceitos para atingir esses objetivos, ou seja, qual a missão do produto multidisciplinar.
Ocorrência: Ao final da fase de Definição de Expectativas
Critério de sucesso: Necessidades dos <i>Stakeholders</i> devem ser esclarecidas e documentadas e a proposta deve demonstrar que pode as solucionar.
Saída: Elicitação de necessidades e objetivos dos <i>Stakeholders</i> aprovada para seguir para a elaboração de requisitos.
RIR – Revisão Integrada de Requisitos
Conteúdo: Lista contendo descrição, razão e método de verificação de requisitos funcionais e de desempenho para o produto multidisciplinar e suas interfaces.
Ocorrência: Ao final da fase de Definição de Requisitos.
Critério de sucesso: Os requisitos do produto multidisciplinar devem estar claros, concisos e verificáveis.
Saída: Lista de requisitos aprovada para atribuição de funções e elementos.
RIA – Revisão Integrada de Arquitetura

Conteúdo: Modelagem de arquitetura do produto multidisciplinar, contendo a modelagem dos produtos das DEs, interfaces internas e externas.
Ocorrência: Ao final da fase de Definição de Arquitetura.
Critério de sucesso: A arquitetura proposta deve estar de acordo com os requisitos, os quais devem estar alocados para todos os elementos funcionais do produto multidisciplinar.
Saída: Arquitetura do produto multidisciplinar aprovada para implementação de conceitos.
RIPC – Revisão Integrada Preliminar de Conceitos
Conteúdo: Apresentação de forma escrita ou gráfica do conceito multidisciplinar do produto.
Ocorrência: Na metade da fase de Projetação, após as Revisões de Conceito do Produto.
Critério de sucesso: O conceito proposto deve demonstrar que cumpre todos os requisitos de sistema dentro de um risco, custo e cronograma aceitáveis e estabelece a base para evolução do mesmo.
Saída: Conceito(s) inicial(s) do produto multidisciplinar selecionado(s) e aprovado(s) para evolução e definição de componentes.
RICC – Revisão Integrada Crítica de Conceitos
Conteúdo: Apresentação da evolução dos conceitos mecânico, eletrônico, de software e interfaces internas e externas.
Ocorrência: Ao final da fase de Projetação
Critério de sucesso: A documentação deve demonstrar que a maturidade dos produtos é adequada e integrada para prosseguir com a realização dos mesmos.
Saída: Conceito(s) final(is) do produto multidisciplinar selecionado(s) e aprovado(s) para implementação e realização.
RIRP – Revisão Integrada de Realização de Produtos
Conteúdo: Checklist de produção e implementação de produtos, interfaces, produtos de apoio e mão de obra.
Ocorrência: Ao final da fase de Realização
Critério de sucesso: O checklist deve assegurar que todos os produtos, seus segmentos, componentes e produtos de apoio para integração no produto multidisciplinar, assim como infraestrutura de integração, mão de obra de suporte e planos e procedimentos estão disponíveis.
Saída: Aprovação para iniciar a integração dos produtos das DEs.
RIPM – Revisão Integrada do Produto Multidisciplinar
Conteúdo: Relatório de integração e testes preliminares do produto

multidisciplinar.
Ocorrência: Ao final da fase de Integração.
Critério de sucesso: Os produtos realizados devem se integrar de acordo com a arquitetura proposta.
Saída: Produto multidisciplinar integrado aprovado para verificação formal de requisitos.
RVMP – Revisão de Verificação do Produto Multidisciplinar
Conteúdo: Relatório de verificação de requisitos por análise, demonstração, inspeção ou testes.
Ocorrência: Ao final da fase de verificação
Critério de sucesso: Os produtos de DE e produto multidisciplinar devem cumprir os requisitos estabelecidos.
Saída: Produto multidisciplinar aprovado de acordo com os requisitos.
RAPM – Revisão de Aceitação do Produto Multidisciplinar
Conteúdo: Relatório de validação e Aceitação do cliente.
Ocorrência: Ao final da fase de Validação.
Critério de sucesso: O produto multidisciplinar a deve cumprir as necessidades expressas pelos stakeholders. Pode se considerar pronto para o lançamento.
Saída: Produto multidisciplinar aceito pelos <i>Stakeholders</i> .

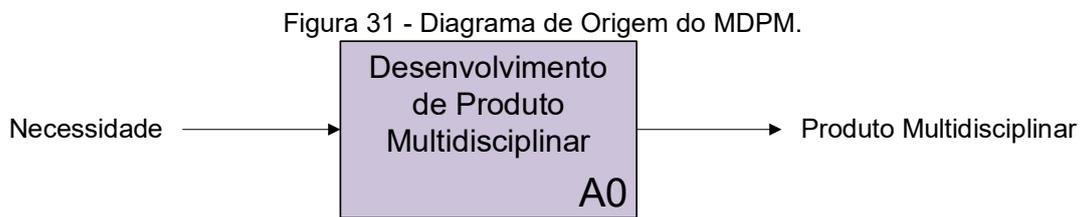
Fonte: o autor.

A condução das revisões integradas se dá pelo engenheiro de sistemas, acompanhado pelos líderes das DEs e o resultado necessita ser aprovado pelo cliente. A continuidade do projeto, em cada etapa de revisão integrada, se dá por sua aprovação. Os *templates* propostos para as revisões integradas estão disponíveis no Apêndice C.

3.2.2.3 Processos do MDPM

De forma a detalhar o processo de desenvolvimento de cada DE, acompanhando o ciclo de desenvolvimento proposto no modelo, MDPM pode ser representado de uma maneira *top-down*, do geral aos detalhes. A ferramenta escolhida para essa representação é o IDEF0, próprio para modelagem de processos (Calil, 2008). A Figura 31 apresenta o diagrama de origem do MDPM, no qual uma necessidade de *stakeholders* pode se tornar

um produto multidisciplinar através de uma função principal que é o “Desenvolvimento de Produto Multidisciplinar”.



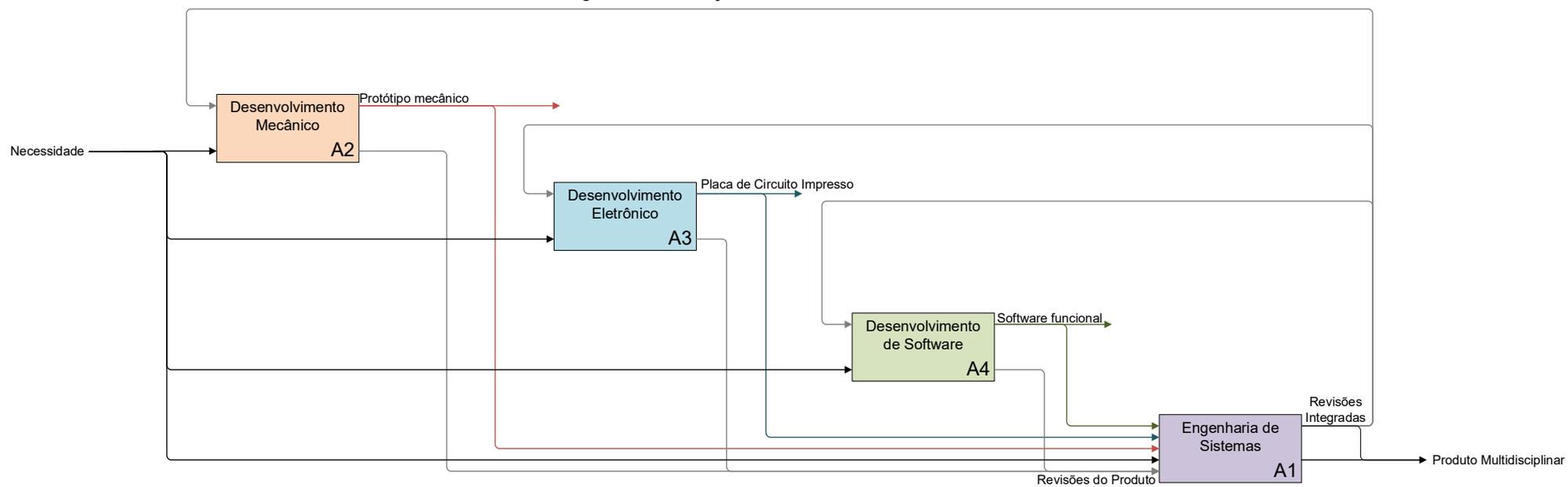
Fonte: o autor.

As relações entre DEs e ES podem ser visualizadas no próximo nível, conforme a Figura 32, onde se diferenciam os processos de desenvolvimento das DEs e a própria ES. Todos os blocos recebem como entrada a necessidade expressa pelo *stakeholder* e como mecanismo as DEs. A ES se estabelece como controle em todos os processos, aliada aos controles específicos para cada DE:

- PDP para o desenvolvimento mecânico - representando mecanismos do modelo PRODIP;
- PDE (e PDP para as primeiras fases) para o desenvolvimento eletrônico;
- Ágil (o termo é utilizado representando práticas de modelos ágeis) para o desenvolvimento de software.

Nas setas de saída dos blocos indica-se a recursividade do modelo: cada DE tem como saída seu produto, bem como as revisões desse; a composição das saídas de cada DE servem como entrada para a ES; a ES tem como saídas as revisões integradas, que retornam como informação de entrada para as DEs, buscando-se agregar um sistema – um produto multidisciplinar. Apesar de se evidenciar os processos de forma individual, conforme a Figura 32, esses desenvolvimentos ocorrem em paralelo.

Figura 32 - Relação entre DEs e ES.



Fonte: o autor.

A seguir, são apresentados os detalhes de cada bloco no segundo nível de desdobramento. A ES é o primeiro diagrama desdobrado (terceiro nível) apresentado, e sua atuação ocorre em todas as fases no modelo. Conforme a Figura 33²², o fluxo de transformações (entradas e saídas) das fases se inicia com uma necessidade e percorre todas as fases até se tornar um produto multidisciplinar. Além desse fluxo, há o mencionada recursividade de modelo: as revisões do produto (RRP, RAP, RCP e RTP) são entradas nas fases de domínio das DEs; as revisões integradas (RIO, RIR, RIA, RIPC, RICC, RIRP, RIPM, RVPM e RAPM) são saídas em todas as fases. Como será visto adiante, a ordem se inverte nos processos das DEs.

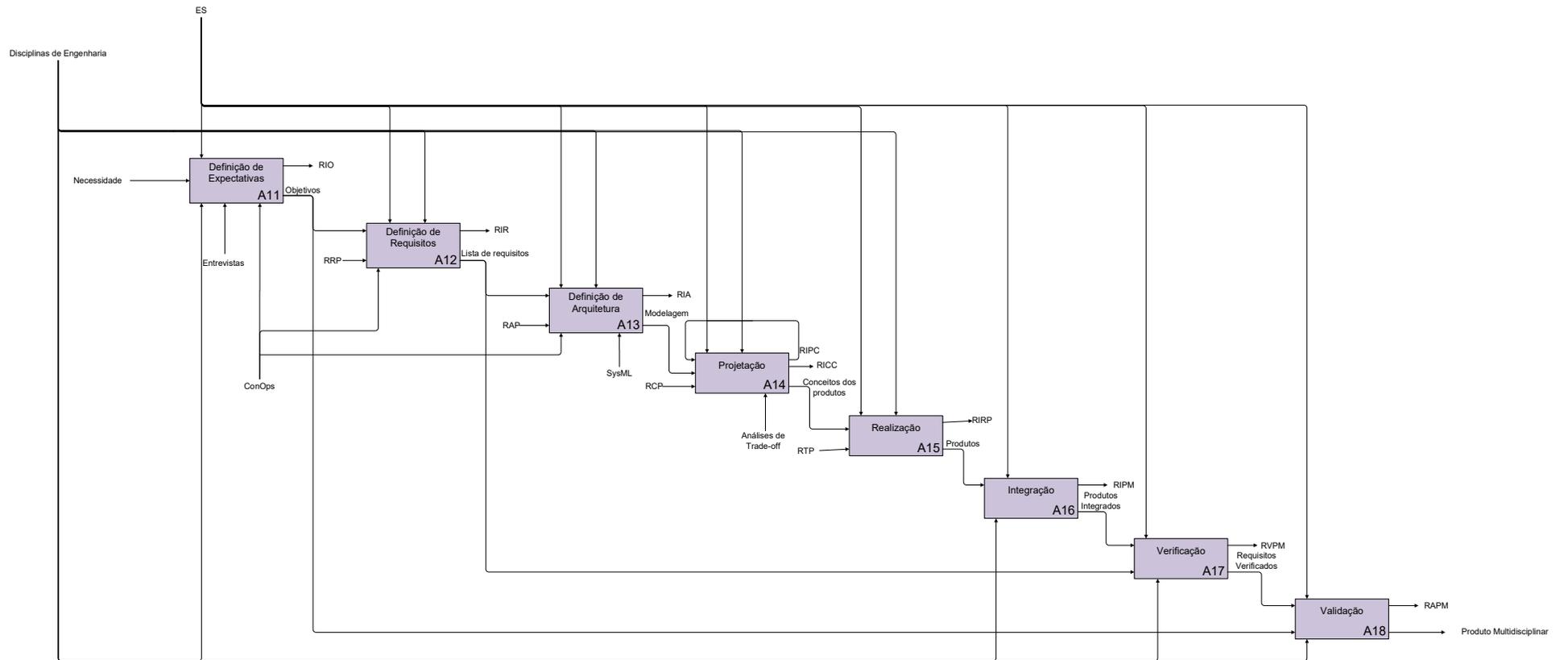
Observa-se também na Figura 33 a indicação de controles (*i.e.* normas ou procedimentos) e mecanismos de execução (*i.e.* ferramentas, métodos e/ou diretrizes de suporte). A ES atua como controle durante todo o processo, enquanto as DEs atuam ora como controle, ora como mecanismo, de acordo com o domínio da fase. Demais mecanismos são provenientes de modelos de ES: o ConOps, com o intuito de favorecer o entendimento e a decomposição do problema; a linguagem SysML (apresentada na seção 2.1.4.3), a fim de prever a integração das DEs desde as primeiras fases, e as análises de *Trade-off*, que auxiliam na escolha de soluções.

A última saída do diagrama é a entrega principal do MDPM, o produto multidisciplinar.

A seguir são representados os processos das DEs. O estabelecimento do desenvolvimento de cada DE se baseia no Quadro 2, ao se verificar a relação de fases do MDPM e as dos modelos-base, e no Quadro 1. Diferentemente da ES, o segundo nível de desdobramento das DEs mostra somente aquelas fases que são de domínio das DEs.

²² Para melhorar a legibilidade do bloco, algumas palavras nos títulos das fases foram omitidas.

Figura 33 - Atividades da Engenharia de Sistemas.



Fonte: o autor.

O processo de desenvolvimento mecânico proposto pelo MDPM pode ser visualizado na Figura 34.

O fluxo de transformação do desenvolvimento mecânico se baseia nos controles principais: a ES e o PDP. Além desses, sugere-se controles próprios de acordo com a fase do ciclo de desenvolvimento: componentes COTS (*Commercial-off-the-shelf* ou componentes padronizados “de prateleira”, ou seja, componentes comerciais), normas aplicáveis de acordo com o ambiente do produto e os métodos de fabricação do mesmo.

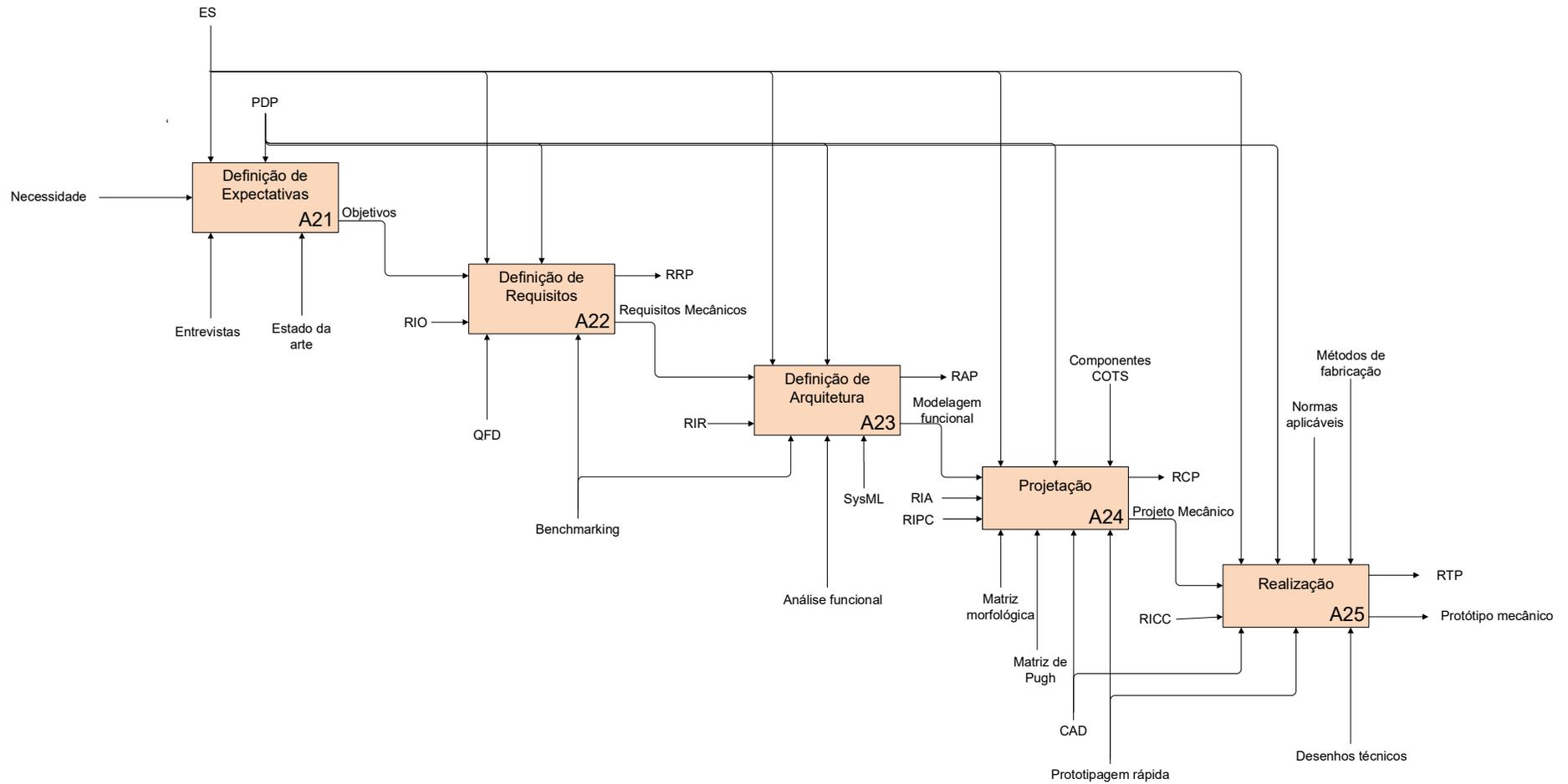
Os mecanismos têm, em sua maioria, origem nos modelos PDP propostos por Pahl *et al.* (2007) e PRODIP, proposto por Back *et al.* (2008). Alguns mecanismos não aparecem de forma explícita nos modelos-base:

- A linguagem de modelos SysML, advém da ES para a modelagem de sistemas. É proposta a fim de facilitar a integração com outras DEs;
- As provas de conceito e prototipagem rápida são mecanismos observados na pesquisa exploratória (Apêndice A) para auxiliar na projeção e realização dos produtos.

Observa-se novamente a recursividade das revisões propostas: as revisões integradas (RIO, RIR, RIA, RIPC, RICC) atuam como entrada a partir da fase de Definição de Requisitos e as revisões do produto (RRP, RAP, RCP e RTP) como saída, permitindo o desenvolvimento integrado com outras DEs.

Ao final do processo se estabelece o desenvolvimento do protótipo mecânico, o qual será integrado com demais produtos das DEs.

Figura 34 - Atividades de Desenvolvimento Mecânico.



Fonte: o autor.

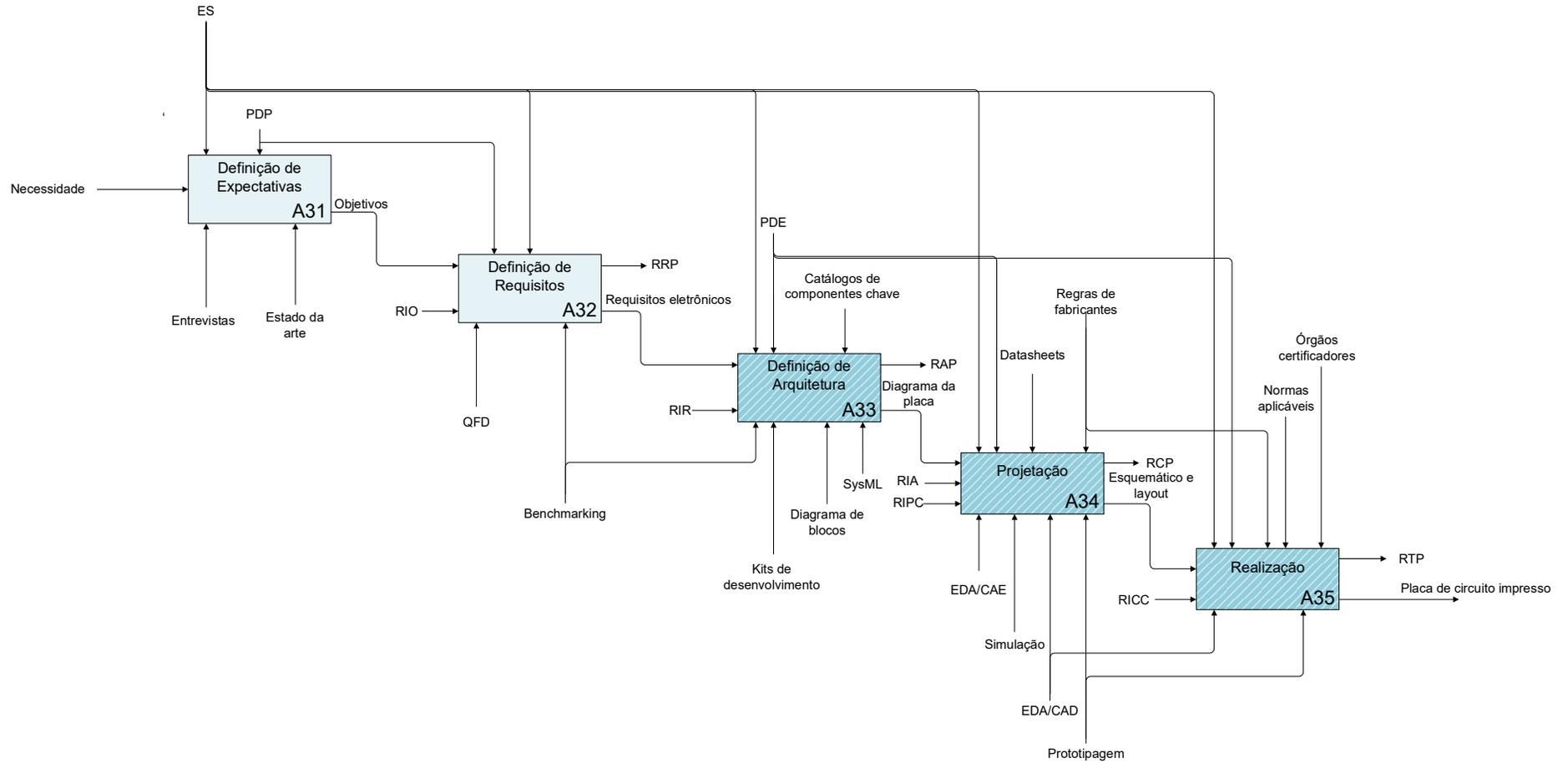
De maneira semelhante, o processo proposto para o desenvolvimento eletrônico pode ser visto na Figura 35. A diferença de tonalidades nos blocos destaca fases encontradas na literatura (com hachuras), com controle do PDE, e fases não contempladas nas fontes pesquisadas (em azul claro), com controle do PDP.

A ES atua como controle em todo o processo novamente. Além dos mencionados, há controles para fases específicas, tais como catálogos para escolha dos componentes principais, regras de fabricantes de placas para direcionar algumas decisões de projeto, normas aplicáveis, e, se necessário, órgãos certificadores.

Os mecanismos são provenientes do levantamento da literatura do PDE e alguns advindos da pesquisa exploratória, como a realização de provas de conceito com *kits* de desenvolvimento para auxiliar a definir a arquitetura. Assim como no desenvolvimento mecânico, propõe-se que a modelagem em diagrama de blocos seja adaptada posteriormente ao SysML para favorecer a integração prévia com as outras DEs.

A saída principal compreende uma placa de circuito impresso e, a cada fase, as revisões do produto. Da mesma maneira que no desenvolvimento mecânico, as revisões integradas atuam como informações de entrada do processo, de forma recursiva.

Figura 35 - Atividades de Desenvolvimento Eletrônico.



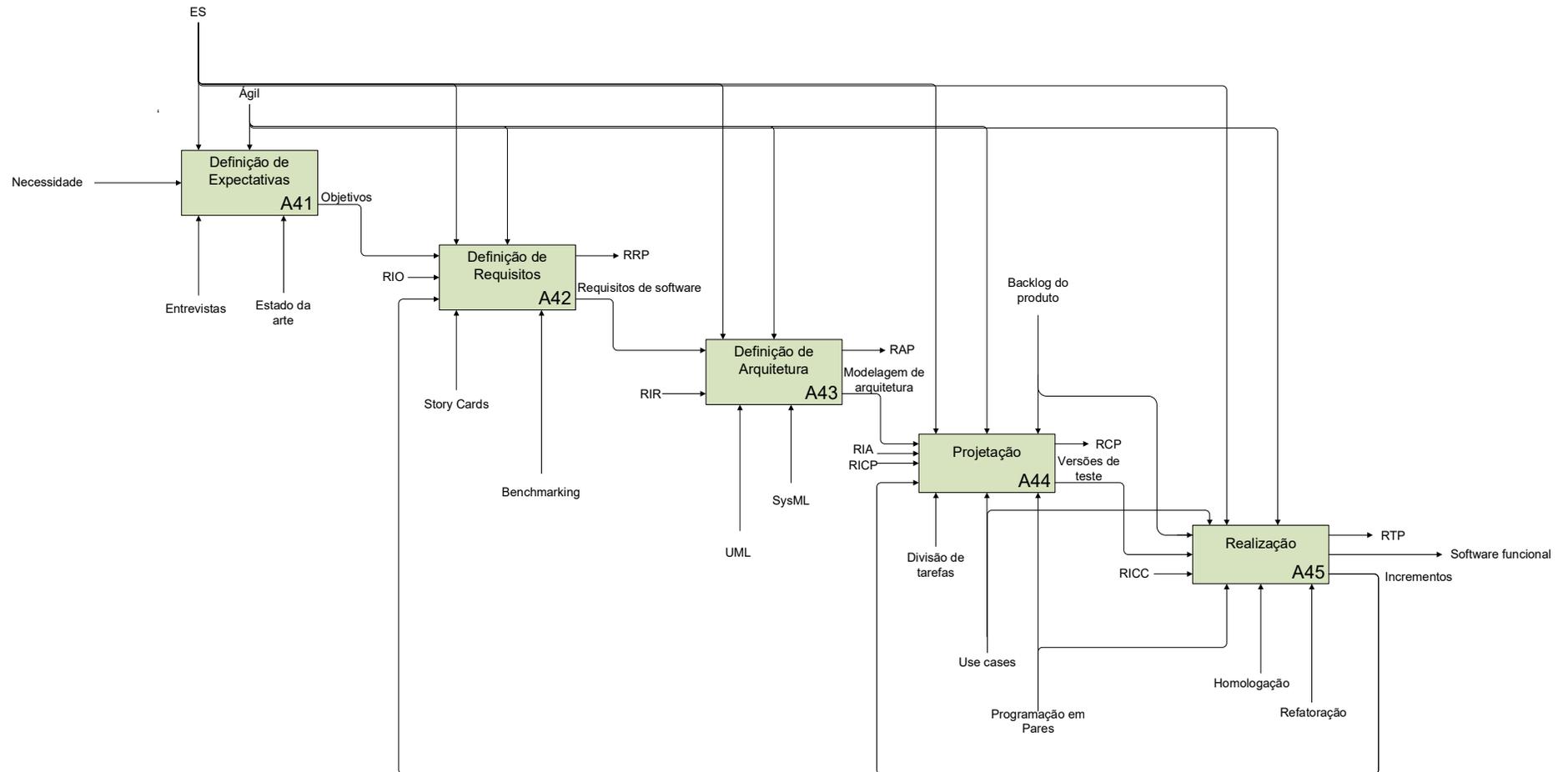
Fonte: o autor.

O desenvolvimento de software, apresentado na Figura 36, tem uma particularidade em relação aos anteriores, apresentando ciclos mais curtos de desenvolvimento. Devido à natureza recursiva e iterativa desse processo, as saídas da fase de Realização e Implementação podem ser incrementais, retroalimentando tanto os requisitos, quanto a fase de projeção, ou ainda compondo versões de um software funcional.

Assim como nas outras DEs, a ES atua como controle durante todo o processo, tal qual o Ágil (representando os modelos ágeis). Os mecanismos propostos derivam de modelos ágeis (especialmente no modelo XP), como os *storycards* para levantamento e priorização de requisitos, a divisão de tarefas e os *use cases*, e práticas como programação em pares e refatoração. A arquitetura é modelada com a linguagem UML e, novamente, a linguagem SysML é recomendada para proporcionar análise prévia de integração entre as DEs.

O processo apresenta revisões integradas como informação de entrada e revisões do produto como saídas. Além dos incrementos de software a cada ciclo de desenvolvimento, a saída principal do processo de desenvolvimento é um software funcional.

Figura 36 - Atividades de Desenvolvimento de Software.



Fonte: o autor.

3.3 CONSIDERAÇÕES DO CAPÍTULO

O presente capítulo apresentou o Modelo de Desenvolvimento de Produto Multidisciplinar. Os processos, fases e mecanismos propostos buscam alinhar boas práticas da engenharia de sistemas e alguns dos modelos de desenvolvimento mais difundidos na academia, juntamente com práticas observadas na pesquisa exploratória.

Esses processos têm um fluxo esperado, no qual, a partir de uma necessidade, chega-se até o produto multidisciplinar desenvolvido. No entanto, entende-se que diversas etapas ou atividades no processo de desenvolvimento de novos produtos são iterativas, incrementais ou ainda necessitam ser revisitadas. Ainda, em se tratando de inovação (incertezas), desvios do fluxo esperado podem ocorrer, ocasionando retrabalho. Para complementar a discussão do modelo proposto, o Apêndice D representa os fluxos esperados de recursividade e potenciais fluxos relacionados a retrabalho.

Uma vez proposto, o modelo deve ser avaliado conforme sua aplicação. O próximo capítulo mostra a aplicação do MDPM, assim como a avaliação do modelo em 3 distintos contextos.

4 APLICAÇÃO E AVALIAÇÃO DO MDPM

O presente capítulo apresenta a avaliação do MDPM conforme três contextos: uma aplicação durante a execução de um projeto de P&D, com posterior avaliação por parte dos participantes; uma avaliação conceitual por parte de desenvolvedores especialistas atuantes na iniciativa privada; uma avaliação conceitual por parte de especialistas pesquisadores. As seções a seguir tratam de cada um desses contextos, respectivamente.

4.1 APLICAÇÃO EM PROJETO DE P&D

O modelo foi aplicado parcialmente²³ em um projeto de P&D em execução, no âmbito do Edital da ANEEL para desenvolvimento de tecnologias e soluções para o setor elétrico. Esse projeto teve início no ano de 2023 e sua finalização está prevista para o final do ano de 2025.

O projeto tem como objetivo o desenvolvimento de uma tecnologia, composta por um dispositivo físico, com eletrônica embarcada e software, destinado à problemática de compartilhamento de infraestrutura, ou seja, o uso de postes de distribuição para passagem de cabeamento de telecomunicações. Por questões de sigilo e confidencialidade, informações mais detalhadas sobre o desenvolvimento não podem ser apresentadas.

A equipe executora, no momento da aplicação do modelo, estava dividida conforme as seguintes subequipes para realização das tarefas específicas:

- Mecânica: equipe responsável pelo desenvolvimento da parte mecânica e estrutural do dispositivo;
- Eletrônica: equipe responsável pelo monitoramento, acionamento, registro e comunicação do dispositivo e usuários;
- Software: equipe responsável pelo tratamento de dados e pelo desenvolvimento da interface com usuários, ou seja, identificação, monitoramento e disponibilização de informações;
- Gestão: equipe responsável pelo planejamento e acompanhamento da execução das atividades previstas, registro e acompanhamento físico-financeiro do projeto, realização de orçamentos e memorandos de aquisições.

²³ O projeto já estava em desenvolvimento com o modelo PRODIP. A aplicação do MDPM se deu de forma paralela durante o tempo disponível.

As subequipes se alinham à proposta das DEs (Desenvolvimento mecânico, desenvolvimento eletrônico e desenvolvimento de software), enquanto as responsabilidades da equipe de gestão, principalmente no que tange ao planejamento e acompanhamento da execução das atividades previstas, visando a integração da solução como um sistema ou produto final funcional, condizem com parte do escopo da ES.

No momento da aplicação o projeto já se encontrava em execução. Dessa maneira, para adequar o início da aplicação, é importante esclarecer o estágio do projeto quando a aplicação ocorreu.

4.1.1 Estágio do projeto no momento da aplicação

O projeto é desenvolvido utilizando como base o modelo PRODIP (Figura 7). Entende-se que a macrofase de planejamento foi realizada juntamente com a proposta do projeto (*i.e.* submissão), com as atividades de busca de anterioridade em patentes, artigos científicos e soluções comerciais.

O projeto informacional já havia sido encerrado no momento da aplicação, tratando da identificação e priorização de necessidades, requisitos e explicitação de restrições impostas por legislações ou pela instituição contratante.

A fase em execução, conforme o modelo originalmente adotado pela equipe, era o projeto conceitual. A equipe de mecânica desenvolveu 5 conceitos, os quais foram apresentados, analisados e priorizados pelo cliente. A equipe de eletrônica realizou testes de componentes para comunicação e se encontrava na definição de sensores, processadores e fonte de alimentação. A equipe de software desenvolveu uma versão inicial do banco de dados, interfaces e uma primeira versão para testes de usabilidade com dados fictícios.

Dessa maneira, a aplicação do MDPM se deu durante a fase de “Projetação”. O autor realizou, de maneira presencial, uma apresentação prévia de toda a proposta e estrutura do modelo à equipe do projeto. Diretrizes propostas pelo modelo foram sugeridas e usadas como suporte para implementação e acompanhamento no projeto. O pesquisador responsável pela coordenação e gestão do projeto foi selecionado para atuar como o engenheiro de sistemas, e instruído sobre o modelo, os entregáveis e as revisões previstas durante essa fase.

Condizente com a fase de aplicação, foram disponibilizados *templates* para a revisão do produto RCP e as revisões integradas RIPC e RICC. De acordo com o

recomendado no MDPM, os líderes de DE conduziram reuniões e revisões do produto, que foram analisadas e aprovadas pelo engenheiro de sistemas, com recomendações a serem implementadas previamente às revisões integradas. Essas foram conduzidas pelo próprio engenheiro de sistemas, constando recomendações também advindas do cliente. Ressalta-se que, durante todo o período da aplicação, o autor esteve presente como observador, não atuando em nenhuma equipe de DE ou ES – somente sanando dúvidas em relação aos entregáveis e revisões solicitadas.

4.2 AVALIAÇÃO

4.2.1 Contexto de projeto P&D

A equipe do projeto é composta por professores e discentes (de graduação e pós-graduação) da Universidade Federal de Santa Catarina. Cada subequipe tem a supervisão de professores com domínio de conhecimento na área de desenvolvimento. A composição vigente das equipes de execução era:

- equipe de engenharia de sistemas: composta por um professor doutor e uma pós-doutoranda;
- equipe de desenvolvimento mecânico: quatro bolsistas graduandos de engenharia mecânica e dois professores doutores;
- equipe de desenvolvimento eletrônico: um bolsista mestrando de engenharia mecânica, dois bolsistas graduandos de engenharia elétrica e um professor doutor;
- equipe de desenvolvimento de software: um doutorando em engenharia mecânica, um graduando em sistemas de informação e um professor doutor.

Por serem os responsáveis pela execução das atividades, os discentes foram as pessoas consultadas posteriormente para realizar a avaliação do modelo. Após a aplicação, foi solicitado que os executores avaliassem a aplicação do modelo através de um formulário disponibilizado de forma *on-line*, estruturado conforme escala Likert, qualitativa, bem como algumas questões abertas foram propostas de forma a obter percepções e sugestões. Dos 9 discentes envolvidos, 7 responderam ao formulário, conforme a Tabela 1.

Tabela 1 -Perfil da equipe técnica avaliadora em contexto de Projeto P&D.

Formação	Equipe		
	Mecânica	Eletrônica	Software
Graduando	3	2	
Mestrando		1	
Doutorando			1

Fonte: o autor.

O Quadro 5 apresenta o questionário de avaliação. O questionário em contexto de projeto P&D teve como intenção avaliar a experiência da utilização do MDPM em projeto, mesmo que de maneira parcial.

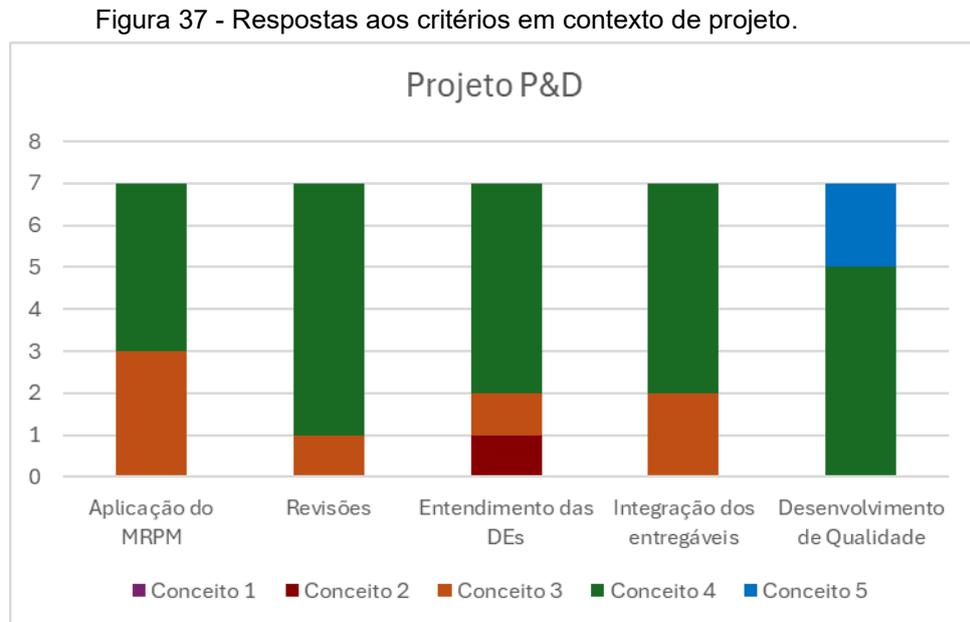
Quadro 5 - Questionário de avaliação por equipe de projeto

De acordo com sua experiência na aplicação do MRM:				
1) O modelo de referência apresenta fácil entendimento para a aplicação?				
Não	Pouco	Razoavelmente	Sim	Completamente
2) Considerando a fase atual de projeto, a aplicação das revisões sugeridas (PCR, PDR e CDR) pelo modelo de referência ocorreu de forma orgânica?				
Não	Pouco	Razoavelmente	Sim	Completamente
3) Considerando a fase atual de projeto, as revisões sugeridas pelo modelo contribuíram para o seu entendimento das outras DEs e do impacto de sua DE nas demais?				
Não	Pouco	Razoavelmente	Sim	Completamente
4) As discussões geradas pelas revisões sugeridas contribuíram para que os entregáveis da sua DE estivessem compatíveis para integração com aqueles das outras DEs?				
Não	Pouco	Razoavelmente	Sim	Completamente
5) Você estima que a utilização do modelo proposto auxilia no desenvolvimento de um produto multidisciplinar de qualidade?				
Discordo totalmente	Discordo	Indiferente	Concordo	Concordo plenamente

Fonte: o autor.

De forma a tornar as respostas mais compreensivas, se atribuíram, para a análise de resultados, valores de 1 a 5 à escala Likert, nos quais 1 é a atribuição mais fraca (representada por “Não” ou “Discordo totalmente”) e 5 é a atribuição mais forte (representada por “Completamente” ou “Concordo plenamente”). Apesar de não haver uma relevância estatística, utilizar os valores auxilia a compreensão da

avaliação. As respostas a cada questão podem ser visualizadas no gráfico de barras da Figura 37, onde o eixo y (vertical à esquerda) mostra a quantidade de respostas, as cores dizem respeito à distribuição das respostas e o eixo x (horizontal) distribui cada pergunta do questionário.



Fonte: o autor.

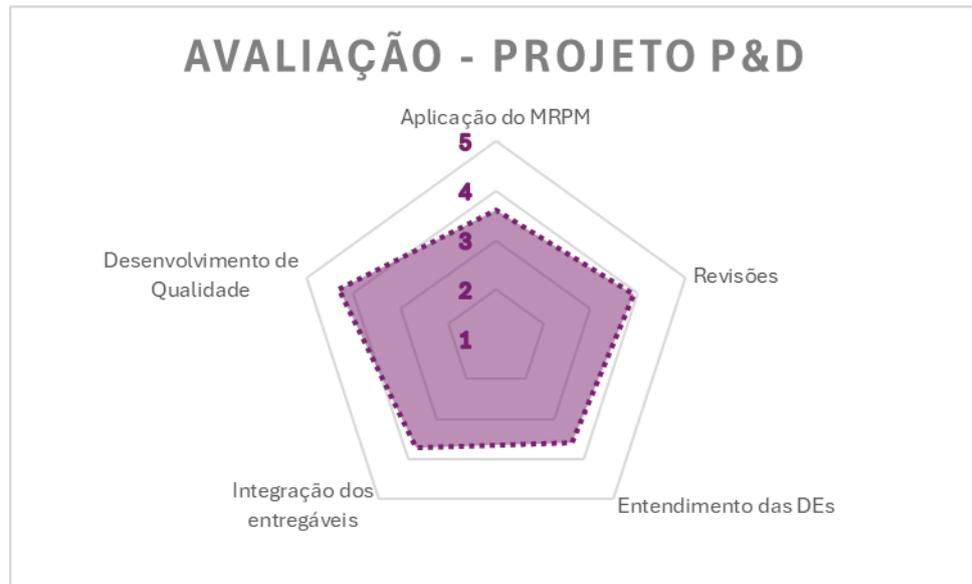
Diante da Figura 37, pode-se observar uma predominância das respostas nos conceitos 3 e 4, o que pode atribuir que:

- A avaliação da questão 1 (com 4 respostas no conceito 4 e 3 respostas no conceito 3) indica um entendimento adequado do MDPM previamente à sua aplicação;
- A avaliação da avaliação da questão 2 indica que as revisões propostas durante o tempo de aplicação foram facilmente incorporadas ao desenvolvimento;
- A avaliação da questão 3 indica que a utilização das revisões sugeridas favorece um maior entendimento sobre o desenvolvimento por partes das equipes de DEs envolvidas;
- A avaliação da questão 4 indica que o MDPM contribui para uma visão multidisciplinar da solução, favorecendo assim a integração durante o desenvolvimento;

- A avaliação da questão 5 indica que a utilização do MDPM tende a beneficiar as equipes a entregarem um produto multidisciplinar de qualidade.

De uma maneira mais macro, o gráfico radar mostrado na Figura 38, mostra que avaliação da aplicação do MDPM de forma geral pode ser considerado de boa aplicabilidade.

Figura 38 - Avaliação do MPDM em contexto de projeto.



Fonte: o autor.

No entanto, há pontos passíveis de melhoria. Dessa maneira, a fim de obter mais informações acerca da aplicação, também foram atribuídas duas questões abertas, agregadas no Quadro 6.

Quadro 6 - Questões abertas da equipe de projeto.

6) O que você identificou como novidade e/ou benefício na aplicação do modelo proposto para o processo de desenvolvimento?	
Resposta 1	<i>“Ajuda a todas as áreas a terem um bom entendimento das outras, porém sem sobrecarregar de informações.”</i>
Resposta 2	<i>“Manter um desenvolvimento coeso com todas as DEs, para que assim não ocorra problemas de integração durante o desenvolvimento do projeto.”</i>
Resposta 3	<i>“O modelo formaliza como o processo de desenvolvimento deve ser executado, oferecendo uma estrutura clara por meio de relatórios padronizados para cada equipe, bem como utiliza esses entregáveis para definir diretrizes e ações. A representação visual com as “cordas” representando distintas DEs equipe ficou bastante intuitiva.”</i>
Resposta 4	<i>“A forma como as integrações são realizadas nas equipes, a</i>

	<i>gestão da divisão de trabalho e a maneira como as atividades são reorganizadas para a colaboração ao final de cada etapa do processo, com isso pode apresentar melhoria no fluxo de trabalho, deixando-o eficiente e coeso.”</i>
7) Você possui alguma sugestão de alteração no modelo proposto ou em sua aplicação?	
Resposta 1	<p><i>“Todas as siglas devem ser definidas nos documentos. Por exemplo, sei o que é “DE” pelas apresentações e e-mails, mas essa definição não está no template. Colocar primeiro o nome completo e depois a sigla, e.g., Disciplina de Engenharia (DE). PCR - Revisão de Conceito do produto; por que não RCP se está em português²⁴?</i></p> <p><i>Esclarecer quem é responsável por preencher cada parte do documento. Por exemplo, a seção “Apresentação do Conteúdo” deve ser preenchida por membros da DE; “Recomendações e Ações necessárias” pelo Engenheiro de Sistemas, etc.</i></p> <p><i>O termo “Data:” no início do documento refere-se à deadline de entrega do documento? Recomendo especificar as datas de cada fase, como: início da fase (ex.: definição da arquitetura), término da fase, prazo para preenchimento do documento, etc.</i></p> <p><i>É importante definir com clareza quando/ a partir de quais entregáveis cada fase começa e termina. Por exemplo, em termos de entrega de software não ficou claro onde a fase de “projeção” termina e a de “realização” começa. As versões do software que já tiveram o deploy realizado ainda se enquadrariam como conceitos?</i></p> <p><i>É solicitado “Registro visual de conceito ou protótipos iniciais”, mas acho importante que as DEs expliquem as imagens ou protótipos. Além disso, informações como estrutura de dados não são facilmente capturadas por imagens.</i></p> <p><i>Ajudaria bastante se no template fossem descritos vários exemplos do que é esperado, e.g., para mecânica: modelos CAD, estrutura da BOM, etc;</i></p> <p><i>Os prazos para preencher os documentos devem ser considerados, assim como para analisar as recomendações e ações propostas.</i></p> <p><i>Reforço que estas são apenas sugestões que, a meu ver, podem trazer ainda mais clareza. No entanto, parabéns pelo trabalho!”</i></p>
Resposta 2	<i>“Senti falta da integração do modelo no projeto. Se desde o início do projeto o modelo estivesse claro para a equipe ficaria mais fácil para avaliar a consistência do modelo junto ao projeto.”</i>

Fonte: o autor.

²⁴ Após a aplicação algumas siglas e termos próprios do modelo foram alterados para português. O próprio nome do modelo foi alterado de MRM para MRPM e, finalmente MDPM.

As respostas agregadas no Quadro 6 evidenciam que a proposta do MDPM de melhorar a integração das DEs tende a ser atendida. No entanto, para tornar a aplicação mais orgânica, as informações apresentadas nos *templates* propostos devem estar mais claras, tanto em contextos gerais, quanto em contextos específicos de DEs. Ademais, a boa avaliação da questão 5, aliada à resposta 2 permite inferir que a aplicação do modelo em um tempo maior de projeto P&D poderia proporcionar uma avaliação mais adequada.

4.2.2 Contexto empresarial

No contexto empresarial a equipe técnica acessada foi composta por colaboradores de uma empresa de desenvolvimento em sistemas embarcados. Essa equipe tem experiência em desenvolvimento de produtos multidisciplinares, com os pesquisadores em disciplinas de engenharia: desenvolvimento mecânico, desenvolvimento eletrônico e desenvolvimento de software.

Para avaliação do modelo, este foi apresentado de forma detalhada em reunião específica com cada uma das três equipes de DEs. No total, responderam ao questionário 16 colaboradores. A distribuição dos participantes conforme área DE e nível de graduação é apresentada na Tabela 2.

Tabela 2 - Perfil da equipe técnica avaliadora em contexto empresarial.

Formação	Equipe		
	Mecânica	Eletrônica	Software
Graduação		1	4
Mestrado	5	2	3
Doutorado	1	1	1

Fonte: o autor.

Também foi elicitado, conforme apontado na Tabela 3, o nível de conhecimento prévio dos participantes no que tange aos modelos de desenvolvimento de soluções ou produtos.

Tabela 3 - Conhecimento prévio da equipe avaliadora em modelos de desenvolvimento

Conhecimento prévio	Equipe		
	Mecânica	Eletrônica	Software
Nenhum		1	
Pouco	1	2	1
Razoável	1	1	5
Bom	2		2
Muito bom	2		

Fonte: o autor.

Após a apresentação, foi solicitado aos pesquisadores responderem um questionário estruturado qualitativo. De forma similar ao contexto de projeto P&D, se atribuem valores à escala a fim de auxiliar na compreensão da avaliação. Considerando-se que os pesquisadores da instituição apresentam perfis técnicos semelhantes ao daqueles respondentes na pesquisa exploratória inicial (Apêndice A), o questionário de avaliação no presente contexto procurou indagar se os principais dos pontos levantados – *i.e.* definição de pontos de integração, processos e entregáveis específicos por área e pessoas de referência – são atendidos pelo MDPM. O Quadro 7 apresenta o questionário elaborado e aplicado.

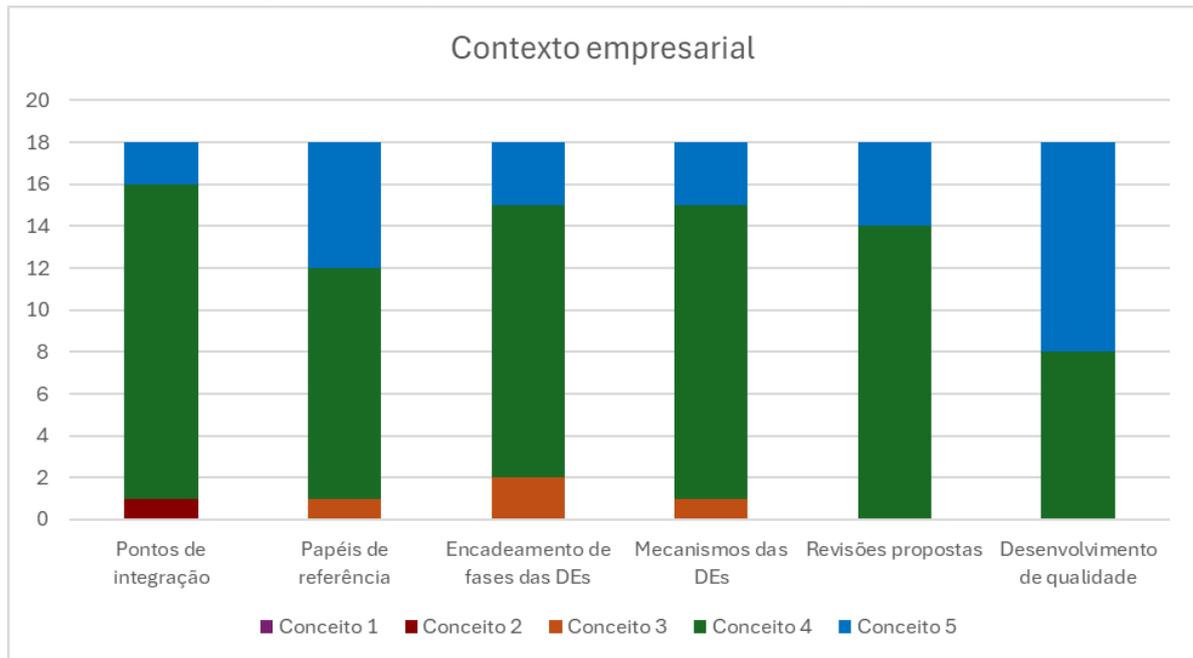
Quadro 7 - Questionário para contexto empresarial.

De acordo com sua experiência no desenvolvimento multidisciplinar de produtos:				
1) Os pontos de integração entre DEs estão adequados?				
Não	Pouco	Razoavelmente	Sim	Completamente
2) Os papéis de referência propostos (líderes de DEs e ES) beneficiam o desenvolvimento multidisciplinar de produto?				
Discordo totalmente	Discordo	Indiferente	Concordo	Concordo plenamente
3) O encadeamento de fases (entradas e saídas) da sua DE é apropriado?				
Não	Pouco	Razoavelmente	Sim	Completamente
4) Os mecanismos (ferramentas e atividades) propostos para execução da sua DE são pertinentes?				
Não	Pouco	Razoavelmente	Sim	Completamente
5) As revisões propostas (de produto e integrada) são pertinentes?				
Não	Pouco	Razoavelmente	Sim	Completamente
6) A utilização do modelo de referência pode beneficiar o desenvolvimento de um produto multidisciplinar de qualidade?				
Discordo totalmente	Discordo	Indiferente	Concordo	Concordo plenamente

Fonte: o autor.

De maneira similar à avaliação em contexto de projeto P&D, foram atribuídos conceitos 1 a 5 à escala Likert para facilitar na interpretação dos dados. A Figura 39 mostra em gráfico de barras as respostas dos entrevistados a cada uma das perguntas do questionário.

Figura 39 - Respostas aos critérios em contexto empresarial.



Fonte: o autor.

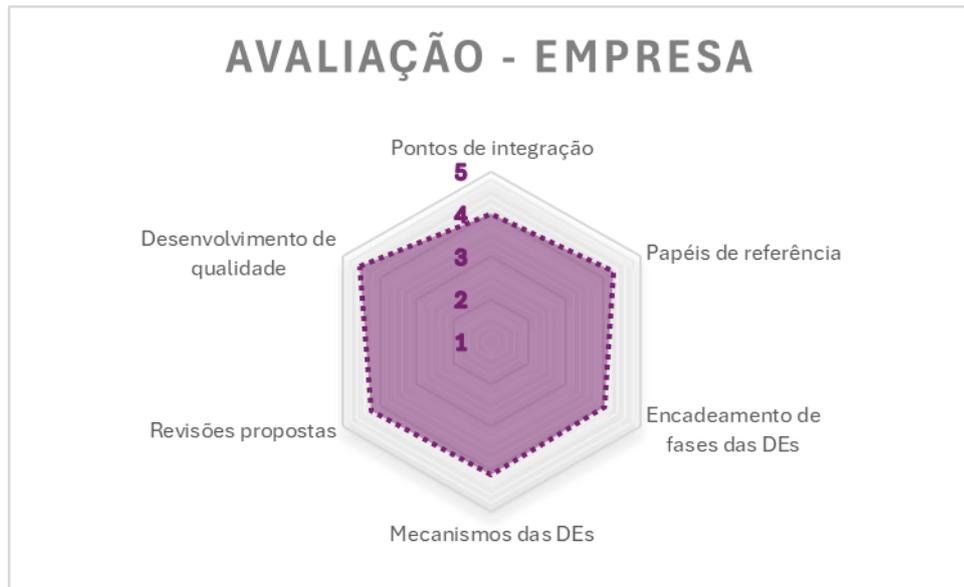
Com uma predominância de respostas nos conceitos 4 e 5 diante dos pontos expostos na pesquisa exploratória (Apêndice A), pode-se considerar que o modelo proposto se adequa às principais sugestões e necessidades levantadas:

- A avaliação da questão 1 indica que o modelo contempla pontos de integração entre as DEs que auxiliam no desenvolvimento multidisciplinar;
- A avaliação da questão 2 indica que os papéis de referência propostos se justificam para beneficiar o desenvolvimento multidisciplinar;
- As avaliações das questões 3 e 4 indicam que a interpretação dos pesquisadores de que o modelo apresenta relevante adequação dos processos específicos para cada DE;
- A avaliação da questão 5 indica que as documentações de aprovação, na forma de revisões, são pertinentes e, de fato, podem auxiliar no desenvolvimento;

- A avaliação da questão 6 indica que a utilização MDPM na empresa poderia trazer benefícios significativos ao desenvolvimento de uma solução de qualidade.

O gráfico radar ilustrado na Figura 40 mostra que o MDPM obteve uma distribuição coerentes em todas as avaliações, apresentando adequação e aderência ao ambiente empresarial.

Figura 40 - Avaliação do MDPM em contexto empresarial.



Fonte: o autor.

Além das questões objetivas, foi solicitado aos avaliadores que fizessem comentários ou sugestões de maneira aberta para o modelo proposto. As respostas estão disponíveis no Quadro 8.

Quadro 8 - Comentários e sugestões da equipe técnica.

Resposta 1	<i>“A proposta é bem robusta e é uma ótima abordagem, considerando que a maioria dos produtos são multidisciplinares e considerando a falta de uma metodologia nesse sentido. Uma sugestão seria lidar com a diferença cronológica entre fases para diferentes áreas.”</i>
Resposta 2	<i>“Na figura do MDPM não acho que na fase Realização você vá gerar um “produto”, mas sim um protótipo. O produto é só lá no final da validação. Na fase de projeção (desenvolvimento mecânico) incluiria um “mecanismo de execução” voltado para simulação dinâmica e simulação estrutural/térmica/fluida (CAE)”</i>
Resposta 3	<i>“Talvez descentralizar ainda mais os meios de comunicação entre as equipes, permitir a livre troca de informação entre integrantes das diferentes áreas.”</i>

Resposta 4	<i>“A partir da apresentação feita, o modelo está bem abrangente e satisfaz as 3 disciplinas que se propõe. Seria necessário testar com a aplicação em algum projeto para identificar eventuais melhorias.”</i>
Resposta 5	<i>“Uma das dores que já senti o realizar projetos multidisciplinares é como realizar "na prática" as integrações entre as diferentes áreas, e ter registrado as expectativas para a integração, e resultados obtidos nesta. Não tenho certeza se isso está no escopo do trabalho, mas ter um detalhamento maior destes pontos de intersecção ajudaria bastante.”</i>
Resposta 6	<i>“O único ponto que fiquei na dúvida foi nas atividades de Desenvolvimento de Software, se o uso de "Use cases" faz sentido na Projetação ou na Definição de Arquitetura A utilização de Use Cases serve tanto para identificar requisitos quanto para validá-los através da definição dos fluxos de interação do sistema com seus atores. Se os Use Cases estão bem definidos isso auxilia na Definição de Arquitetura, por isso a dúvida de onde ele se encaixa no fluxo. No mais, está muito bom o trabalho.”</i>
Resposta 7	<i>“O modelo de referência proposto é muito bem estruturado sendo capaz de beneficiar o desenvolvimento de um produto multidisciplinar. Vale ressaltar, que o modelo consta de métodos diversas áreas de conhecimento (disciplinas) e que nem sempre o primeiro contato é simples. Da forma de como o modelo foi proposta ajuda a estruturar muito bem e atua de forma a internalizar conceitos de diversas áreas. O modelo apresentado é um bom guia de desenvolvimento, e mostra de maneira simples a integração entre as DE.”</i>

Fonte: o autor.

Dos 16 avaliadores, somente 7 responderam às questões abertas. As respostas aos questionários foram anônimas, não sendo possível identificar a DE a qual cada resposta pertence. Como mostra o Quadro 8, os comentários tecidos ao MDPM foram majoritariamente positivos, com algumas sugestões que podem ser analisadas e, potencialmente, incorporadas em pesquisas futuras. Ainda, evidenciam, mesmo que de forma preliminar, a adequação do MDPM para projetos de desenvolvimento de produtos multidisciplinares no contexto empresarial.

4.2.3 Avaliação por parte de Especialistas

Parte relevante da avaliação do modelo também se deu por meio da apresentação e consulta a especialistas em desenvolvimento de produtos. Foram convidados 8 especialistas doutores com perfis em diferentes áreas do

conhecimento, dentre os quais 6 aceitaram participar do estudo. Para cada especialista foi marcada uma reunião *on-line*, com duração média de 40 minutos, na qual foi apresentado o MDPM em detalhes e, posteriormente, foi aplicado questionário de avaliação. O perfil dos especialistas que participaram do processo é descrito no Quadro 9

Quadro 9 - Perfil dos Avaliadores Especialistas.

Avaliador	Ocupação atual
Especialista 1	Engenheiro de Dados em empresa internacional de Desenvolvimento de Software
Especialista 2	Gerente sênior de inovação e indústria 4.0 em empresa multinacional
Especialista 3	Coordenador de pós-graduação em Engenharia de Desenvolvimento de Produtos na Univali
Especialista 4	Professora do departamento de Expressão Gráfica da Universidade Federal de Santa Catarina
Especialista 5	Conselheiro em Engenharia Elétrica/eletrônica e líder de projetos em empresa multinacional
Especialista 6	Professor de Engenharia Mecânica e de Pós-graduação em Engenharia Agrícola na Universidade Federal de Santa Maria

Fonte: o autor.

Após a apresentação do modelo, foi solicitado que os especialistas respondessem um formulário composto com questionário semiestruturado, com questões objetivas estruturadas conforme escala Likert. Esse questionário foi baseado no estudo de Czempler, Bajus e Schäffer (2024) que realizaram uma análise sistemática de literatura para identificar e processar critérios de avaliação de modelos de referência. Os principais critérios identificados pelos autores são²⁵:

- I. AUC (precisão, não-ambiguidade e completeza): as informações do modelo devem ser completas, entendidas sempre da mesma maneira e não possuir ambiguidades;
- II. Relevância: todos os elementos do modelo devem ser relevantes para o objetivo;
- III. Eficiência econômica: a criação e implantação do modelo em empresas deve permitir crescimento e redução de custos em médio a longo prazo;

²⁵ Os autores ainda apresentam o critério de comparabilidade (de metamodelos) que não será adotado nessa dissertação.

- IV. Clareza: a estrutura do modelo deve ter uma representação clara e de fácil leitura;
- V. Estruturação sistemática: os diferentes níveis e camadas do modelo devem resultar em um modelo holístico e coerente;
- VI. Adaptabilidade: o modelo permite ser adaptador e expandido;
- VII. Generalidade: o modelo deve ser genérico o bastante para ser aplicado em diferentes casos.

Os quadros a seguir, do Quadro 10 até o Quadro 16, apresentam as questões elaboradas de acordo com cada critério (de forma agregada).

Quadro 10 - Questionário do critério AUC.

Considerando sua área de competência e experiência no desenvolvimento de produtos:				
Critério – AUC				
1) As fases sugeridas no modelo proposto englobam o ciclo de desenvolvimento necessário para o desenvolvimento multidisciplinar de produtos?				
Discordo totalmente	Discordo	Indiferente	Concordo	Concordo plenamente
2) Os mecanismos apresentados no modelo de referência proposto são suficientes para o desenvolvimento multidisciplinar de produtos?				
Não	Pouco	Razoavelmente	Sim	Completamente
3) A descrição dos mecanismos apresentada no modelo de referência proposto é precisa o suficiente para guiar o desenvolvimento multidisciplinar de produtos?				
Não	Pouco	Razoavelmente	Sim	Completamente

Fonte: o autor.

Quadro 11 - Questionário do critério Relevância.

Critério - Relevância				
4) Os mecanismos apresentados no modelo de referência proposto são necessários e relevantes para o desenvolvimento multidisciplinar de produtos?				
Discordo totalmente	Discordo	Indiferente	Concordo	Concordo plenamente

Fonte: o autor.

Quadro 12 - Questionário do critério Eficiência Econômica.

Critério – Eficiência Econômica				
5) Você estima que a implantação do modelo de referência pode, a longo prazo, favorecer um ganho econômico às empresas no desenvolvimento multidisciplinar de produtos?				
Não	Pouco	Razoavelmente	Sim	Completamente
6) Você estima que os momentos de integração previstos no modelo de referência favorecem reduzir o desperdício atrelado a retrabalho no desenvolvimento multidisciplinar de produtos?				
Não	Pouco	Razoavelmente	Sim	Completamente

Fonte: o autor.

Quadro 13 - Questionário do critério Clareza.

Critério – Clareza				
7) A organização e representação gráfica das fases favorece o entendimento da proposta para o desenvolvimento multidisciplinar de Produto?				
Discordo totalmente	Discordo	Indiferente	Concordo	Concordo plenamente

Fonte: o autor.

Quadro 14 - Questionário do critério Estruturação Sistemática.

Critério – Estruturação Sistemática				
8) O modelo de referência abrange campos de conhecimento necessários para desenvolvimento multidisciplinar de produtos?				
Não	Pouco	Razoavelmente	Sim	Completamente
9) O nível de detalhamento das fases apresentado é suficiente para realizar o desenvolvimento multidisciplinar?				
Não	Pouco	Razoavelmente	Sim	Completamente
10) A organização de mecanismos em fases e integração entre as DEs está apresentada de forma lógica e é coerente com o processo de desenvolvimento multidisciplinar de produtos?				
Discordo totalmente	Discordo	Indiferente	Concordo	Concordo plenamente
11) O fluxograma apresentado proporciona uma visão holística do desenvolvimento multidisciplinar de produtos?				
Não	Pouco	Razoavelmente	Sim	Completamente

Fonte: o autor.

Quadro 15 - Questionário do critério Adaptabilidade.

Critério – Adaptabilidade				
12) O modelo de referência permite ser expandido para agregar novos mecanismos, novas DEs ou áreas de conhecimento?				
Não	Pouco	Razoavelmente	Sim	Completamente

Fonte: o autor.

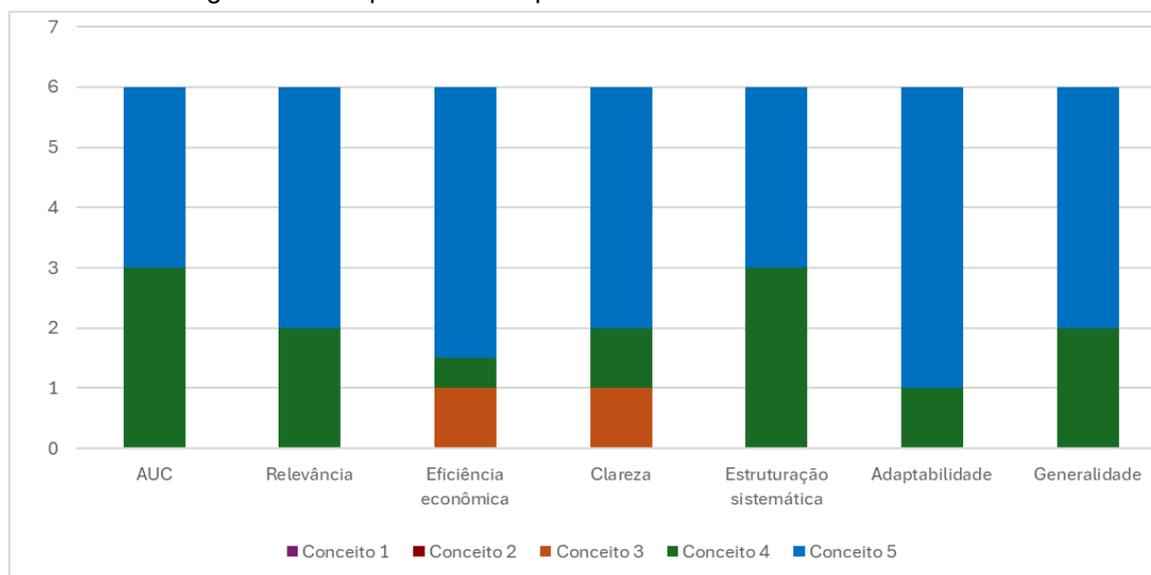
Quadro 16 - Questionário do critério Generalidade.

Critério – Generalidade				
13) O modelo de referência é abrangente o bastante para ser aplicado em diferentes contextos de desenvolvimento?				
Discordo totalmente	Discordo	Indiferente	Concordo	Concordo plenamente

Fonte: o autor.

Assim como nos outros contextos de avaliação, foi feita uma atribuição de conceitos à escala Likert. O gráfico de barras ilustrado na Figura 41 auxilia na visualização da avaliação dos especialistas ao MDPM.

Figura 41 - Respostas dos especialistas aos critérios estabelecidos.



Fonte: o autor.

Diante dos resultados, com quase todos os critérios com predominância dos conceitos 3 e 4, pode-se evidenciar as seguintes interpretações, com base na avaliação de especialistas, a respeito do modelo proposto:

- AUC: o modelo proposto é considerado pelos avaliadores preciso, não-ambíguo e integrativo;
- Relevância: os elementos mostrados no modelo proposto são considerados importantes para o objetivo final;
- Eficiência econômica: os avaliadores entendem que o modelo pode beneficiar economicamente empresas por meio de sua adoção e aplicação;

- Clareza: a representação do modelo é considerada compreensível pelos especialistas;
- Estruturação sistemática: os níveis e processos mostrados para execução do modelo são considerados coerentes de forma a integrarem um modelo bem estruturado;
- Adaptabilidade: o modelo apresenta flexibilidade suficiente para pode agregar novos mecanismos, DEs ou áreas do conhecimento, mantendo a característica de integração, abrangida pela ES;
- Generalidade: o modelo proposto é adequado para ser aplicado em diferentes contextos de desenvolvimento (*i.e.* robótica, sistemas ciber-físicos, produtos *smart*, *retrofit* de máquinas) inseridos do escopo pré-definido.

Assim, pode-se identificar que, de uma maneira geral, o MDPM foi avaliado como adequado e satisfatória pelos especialistas em todos os critérios. O gráfico radar da Figura 42 mostra a distribuição da avaliação por critério.

Figura 42 - Avaliação do MDPM por especialistas.



Fonte: o autor.

Também foi solicitado aos especialistas que preenchessem uma questão aberta com o intuito de tecerem comentários ou sugestões ao modelo. O Quadro 17 apresenta as respostas.

Quadro 17 - Comentários e sugestões dos especialistas ao modelo de referência.

Resposta 1	<i>“Ao final do trabalho gostaria de ter acesso a uma planilha única que resuma todo o modelo proposto e sirva como um guia de aplicação (hands-on).”</i>
Resposta 2	<i>“Para um melhor entendimento de todas as “engenharias” envolvidas no desenvolvimento de produtos multidisciplinares, seria interessante fornecer um dicionário para melhor entendimento de todas as áreas. Seria também interessante linkar as diferentes fases com os níveis TRLs para os projetos de PD&I.”</i>
Resposta 3	<i>“O Modelo de Referência Multidisciplinar para desenvolvimento de produtos me pareceu bastante abrangente, ao incorporar equipes multidisciplinares e combinar abordagens tradicionais e ágeis no desenvolvimento de produtos. De maneira geral, ele inclui as etapas que costumo empregar no processo de desenvolvimento, o que torna o modelo alinhado com práticas já estabelecidas. Contudo, sugiro uma reflexão sobre a forma como a verificação e validação estão estruturadas no modelo. Acredito que estas atividades poderiam ser integradas em cada etapa do ciclo de desenvolvimento, desde a definição de expectativas até a integração, em vez de serem tratadas como fases distintas ao final do processo. Essa abordagem contínua pode garantir maior alinhamento entre as expectativas iniciais e o produto final, reduzindo riscos e melhorando a qualidade ao longo de todo o ciclo de desenvolvimento.”</i>
Resposta 4	<i>“Achei o modelo completo, indicando as fases e mecanismos necessários para a integração. Entretanto, com relação a organização e representação gráfica do processo, percebo que segue uma linguagem de representação técnica (típica da engenharia) que, no futuro, poderia dificultar uma possível integração com outras áreas que também trabalham ou contribuem com o desenvolvimento de produtos (como design de produto, design digital, ux/ui, marketing). Assim como, dificultar a compreensão se prever o uso para o ensino em sala de aula. Como recomendação para trabalhos futuros, uma adaptação para esses outros contextos com um fluxograma mais visual, orgânico e didático poderia auxiliar a compreensão.”</i>
Resposta 5	<i>“O modelo proposto sintetiza bem o processo, cuja natureza é complexa e constitui a possibilidade de realizar diferentes tipologias de projeto com objetivos bastante variados e específicos, o que representa sempre um grande desafio para sua modelagem. Parabéns pela contribuição.”</i>
Resposta 6	<i>“O modelo proposto é interessante, por ser uma tentativa de generalizar o projeto integrado (mecânica, software/firmware, hardware), porém, como</i>

	<p><i>consequência, algumas particularidades não são mencionadas. Por exemplo, no projeto de hardware, as boas práticas de criação de uma biblioteca de componentes, a manutenção desta biblioteca, sua verificação periódica, são fatores importantíssimos (garantir que o footprint correto está sendo usado para cada componente, garantir que os pinos estão alocados da maneira correta, garantir que os pads seguem as dimensões das normas específicas para cada categoria de placa, etc.). As etapas de simulações dos sub-circuitos, em suas mais variadas formas, também vão determinar o sucesso do projeto. As ferramentas de simulação utilizadas, os tipos de simulação. As boas práticas de layout devem ser respeitadas, as simulações de integridade de sinal, para avaliar e evitar cross-talk de trilhas, interferência eletromagnética, susceptibilidade a ruídos ou interferências, etc. As análises de reflexões em casos de sinais de alta frequência, as avaliações dinâmicas (tempo de propagação do sinal, comprimento máximo de trilha) para sinais digitais, etc. A boa comunicação com a empresa que irá realizar a fabricação da placa, e montagem dos componentes, de maneira a garantir que o processo de fabricação é ideal, que os materiais utilizados atenderão as normas/expectativas. A forma como a placa será testada, qualificada etc. Na parte de integração, os diagramas de interface são muito importantes. De alguma forma isso precisa ser explicitamente mencionado. Como sugestão, acredito que seu modelo deveria agrupar mais detalhes de cada disciplina, ao menos como indicações/referências para que os projetistas tenham referências.”</i></p>
--	---

Fonte: o autor.

Assim como nas demais avaliações, as respostas foram enviadas e registradas de forma anônima. Apesar da boa avaliação no critério clareza, os comentários abertos indicam que o MDPM pode se beneficiar de uma representação mais didática, seja por modificações na representação usada, seja pela confecção de um guia de aplicação. Além disso, alguns especialistas apontaram que os momentos de integração, verificação e validação nas próprias DEs, deveriam ficar mais explícitos para redução de riscos durante o desenvolvimento. As sugestões podem ser incorporadas em futuras revisões do modelo.

4.3 CONSIDERAÇÕES DO CAPÍTULO

O presente capítulo apresentou a aplicação e avaliação do MDPM. O contexto de execução em um projeto real de desenvolvimento foi, de certa maneira, limitado devido ao sincronismo temporal entre o projeto e a elaboração da presente dissertação, o que dificultou a aplicação completa do modelo. As sugestões recebidas nas questões abertas mostram que uma das principais justificativas à proposição do modelo, o desenvolvimento paralelo com pontos de integração, foi bem aceita pelos usuários e avaliadores.

De forma complementar, a avaliação em um contexto empresarial buscou ponderar se as necessidades e sugestões levantados na pesquisa exploratória inicial foram de alguma maneira sanados pelo modelo proposto. O resultado para todos os principais pontos levantados foi satisfatório pela opinião dos colaboradores da empresa. Assim, a aplicação do MDPM em um projeto de contexto empresarial no futuro será benéfica para evidenciar vantagens de uma abordagem integrativa estruturada à empresa, bem como proporcionar eventuais refinamentos ao modelo.

A avaliação com especialistas em desenvolvimento de produtos teve um enfoque mais acadêmico, conforme os critérios identificados na literatura. Apesar do MDPM ter sido bem avaliados pelos especialistas em todos os 7 critérios propostos pela literatura, alguns comentários evidenciam pontos de atenção e melhoria para o modelo.

De maneira geral, o MDPM obteve avaliação positiva nos 3 diferentes contextos propostos sendo os comentários apresentados sugestões de suma importância para evolução do modelo.

5 CONCLUSÃO

A presente dissertação teve como intuito colaborar com o processo de desenvolvimento de produtos multidisciplinares. Este capítulo explicita as conclusões gerais da dissertação, as dificuldades encontradas e sugere recomendações para trabalhos futuros.

5.1 CONSIDERAÇÕES E LIMITAÇÕES

Segundo Rozenfeld *et al.* (2006) “o que distingue as empresas com excelência em desenvolvimento de produtos é o padrão de coerência e consistência em todo o processo de desenvolvimento”. Mas, como estabelecer essa coerência se não houver uma linguagem comum e uma visão mínima do andamento do projeto por parte de todos os envolvidos?

Os estudos em metodologia e os modelos dispostos na literatura carregam consigo décadas de conhecimento, profundidade e uma qualidade que os permite ainda hoje serem relevantes. No entanto, o contexto dos produtos se alterou ao longo do tempo, com a incorporação de tecnologias embarcadas e a integração de diferentes áreas do conhecimento, agregando complexidade e tornando o desenvolvimento de produtos dependente de diferentes processos. Para Back *et al.* (2008), desenvolver projetos complexos sem adotar qualquer metodologia ou procedimento é inconcebível.

Nessa conjuntura, foi realizada uma extensa revisão da literatura e uma pesquisa exploratória dentro do contexto de desenvolvimento multidisciplinar. O referencial teórico aponta que cada uma das áreas de conhecimento estudadas tem linguagem, ferramentas e outras particularidades no processo de desenvolvimento, dificultando. A pesquisa exploratória corrobora com esse cenário, no sentido que, segundo pesquisadores, a adoção de um modelo PDP para desenvolvimento eletrônico ou de software, por exemplo, é forçosa e dificultosa. Ademais, o próprio desenvolvimento multidisciplinar carece de uma maior integração e entendimento entres as distintas áreas do conhecimento.

Diante desses levantamentos, foi proposto o Modelo de Desenvolvimento de Produto Multidisciplinar – MDPM. O MDPM busca orientar o desenvolvimento multidisciplinar ao agregar e considerar os processos estabelecidos e particulares de cada área do conhecimento, enquanto provê a integração necessária ao longo dos

ciclos de desenvolvimento, juntamente com as boas práticas da engenharia de sistemas.

A avaliação do MDPM se deu diante de três contextos distintos: a aplicação em um projeto P&D, a avaliação em contexto empresarial e a avaliação por especialistas. A aplicação em projeto foi limitada devido ao tempo de projeto, o que impediu uma exploração e observação mais controlada das atividades e ferramentas sugeridas ao longo da execução do processo. No entanto, a mesma foi importante para o entendimento de que a descrição de alguns processos do MDPM pode ser melhor esclarecida, especialmente diante das especificidades das áreas de conhecimento envolvidas. Já, as avaliações em contexto empresarial e por especialistas contribuíram, respectivamente, para uma visão de profissionais experientes no desenvolvimento multidisciplinar e uma visão técnica acadêmica diante de critérios específicos. O MDPM foi considerado adequado e satisfatório nos três contextos.

Além da avaliação do modelo proposto, é necessário evidenciar o atendimento dos objetivos os quais a presente dissertação se propôs a atingir. Nesse contexto, considera-se que os objetivos foram alcançados:

- **Organizar o conhecimento encontrado na literatura sobre o processo de desenvolvimento de produtos de diferentes áreas de conhecimento:** através de extensa revisão teórica, com embasamento em livros técnicos e artigos científicos relevantes e, também, recentes, conforme apresentado nas seções 2.1, 2.2 e 2.3;
- **Explorar a literatura acerca da engenharia de sistemas como meio de integração de produtos de diferentes áreas do conhecimento:** com embasamento técnico nas principais fontes sobre ES (NASA e INCOSE) e artigos que exploram a atualidade desses assuntos, conforme seção 2.4;
- **Realizar pesquisa exploratória com projetistas de diferentes áreas do conhecimento acerca de seus processos de desenvolvimento, e experiência no desenvolvimento de produto multidisciplinar:** conforme a pesquisa exploratória realizada apresentada no Apêndice A e sua contribuição para a proposição do modelo citada na seção 3.2;
- **Propor um modelo e processo que considerem o desenvolvimento de produtos multidisciplinares:** com a proposição do MPDM na seção 3.2 e a apresentação de seu processo modelado de acordo com o IDEF0;

- **Sugerir atividades, fases e critérios de passagem entre essas, para o desenvolvimento multidisciplinar de produtos:** com a proposição de fases, construídas conforme a pesquisa e apresentada na seção 3.2;
- **Selecionar ferramentas que auxiliem na execução de atividades necessárias para o desenvolvimento de soluções para as específicas áreas do conhecimento:** ao explicitar as atividades e os mecanismos no terceiro nível do diagrama IDEF0, apoiando-se, para isso, no conhecimento levantado no referencial teórico e com auxílio da pesquisa exploratória;
- **Aplicar a proposição do modelo em um caso:** realizado em um projeto P&D apresentado em 4,1;
- **Avaliar o modelo proposto:** com a participação de pesquisadores do projeto P&D, profissionais e especialistas acadêmicos, conforme a seção 4.2.

Dessa maneira, pode-se afirmar que o objetivo geral de “**propor um modelo de desenvolvimento para o desenvolvimento de um produto multidisciplinar capaz de utilizar os processos de desenvolvimento de cada área do conhecimento envolvida garantindo a integração e a compatibilidade entre o desenvolvimento de cada área**” foi atingido.

Entende-se que um próximo passo para o MDPM seja a aplicação em contexto empresarial. Como o modelo é proposto para a execução de DEs com processos bem estabelecidos, entende-se que sua aplicação possa sofrer alguma resistência, sobretudo devido à mudança de cultura requerida na implantação de modelos de desenvolvimento. No entanto, uma vez rompida a barreira cultural, estima-se que a implantação do modelo possa contribuir de forma efetiva para resultados satisfatórios ao desenvolvimento.

Entende-se que os resultados alcançados contribuem para o avanço da resolução dos problemas de pesquisa estabelecidos e que o MDPM se apresenta apto a ser utilizado em novos contextos de projeto de forma integral. Vislumbra-se a adoção do MDPM, suas futuras melhorias e aprimoramentos, em empresas e por estudantes de engenharia que busquem uma visão multidisciplinar para o desenvolvimento de produtos.

Por fim, deve-se agradecer à CELESC, cujo projeto P&D 05697-0423-2023 foi de suma importância na construção e aplicação do modelo proposto na presente dissertação.

5.2 SUGESTÕES PARA TRABALHOS FUTUROS

A fim de aprimorar o modelo, dar robustez aos resultados e continuar com a contribuição para o desenvolvimento multidisciplinar de produtos, recomendam-se como trabalhos futuros:

- Aplicar o MDPM em projetos de diferentes contextos, *i.e.* robótica, produtos *smart* e sistemas ciberfísicos;
- Ponderar as sugestões recebidas nas avaliações para aprimorar o MDPM (e.g. incorporar mais diretrizes e ferramentas de execução e melhorar a legibilidade da apresentação do modelo - didática);
- Propor, de maneira mais estruturada, o desenvolvimento de metodologia ou modelo de referência para o desenvolvimento eletrônico, diante da carência de literatura específica;
- Avaliar a integração de processos de desenvolvimento de outras DEs ao modelo, tais como firmware, instrumentação, controle e automação;
- Criar um guia didático para facilitar a aplicação do modelo em ambiente de projeto empresarial.

REFERÊNCIAS

- ABREU, Cláudia Buhamra. Marketing e produção: ampliando o conceito do produto através dos serviços pós-venda. **ENEGEP**. Gramado, 1997.
- AKBAR, Muhammad Azeem *et al.* Improving the quality of software development process by introducing a new methodology–AZ-model. **IEEE Access**, v. 6, p. 4811-4823, 2017.
- ALMEIDA, Henrique Silveira de; TOLEDO, José Carlos de. Qualidade total do produto. **Production**, [S.L.], v. 2, n. 1, p. 21-37, jun. 1992. FapUNIFESP (SciELO), 1992.
- ALSAQQA, Samar; SAWALHA, Samer; ABDEL-NABI, Heba. Agile Software Development: methodologies and trends. **International Journal Of Interactive Mobile Technologies (Ijim)**, [S.L.], v. 14, n. 11, p. 246, 10 jul. 2020. International Association of Online Engineering (IAOE), 2020.
- AL-TARAWNEH, Mejhem Yousef; ABDULLAH, Mohd Syazwan; ALI, Abdul Bashah Mat. A proposed methodology for establishing software process development improvement for small software development firms. **Procedia Computer Science**, v. 3, p. 893-897, 2011.
- ANWAR, Ashraf. A review of RUP (Rational Unified Process). **International Journal of Software Engineering (IJSE)**, v. 5, n. 2, p. 12-19, 2014.
- ASIMOV, M. **Introdução ao projeto de engenharia**. São Paulo: Mestre Jou, 1968.
- AYDAN, Ufuk *et al.* Teaching ISO/IEC 12207 software lifecycle processes: a serious game approach. **Computer Standards & Interfaces**, v. 54, p. 129-138, 2017.
- BACK, Nelson; OGLIARI, André; DIAS, Acires; SILVA Jonny C.; **Projeto integrado de produtos: planejamento, concepção e modelagem**. Barueri: Malone, 2008.
- BALAJI, Sundramoorthy; MURUGAIYAN, M. Sundararajan. Waterfall vs. V-Model vs. Agile: A comparative study on SDLC. **International Journal of Information Technology and Business Management**, v. 2, n. 1, p. 26-30, 2012.
- BANICA, Logica *et al.* Is DevOps another project management methodology?. **Informatica Economica**, v. 21, n. 3, 2017.
- CLOUTIER, Robert; BALDWIN, Clifton; BONE, Mary Alice. **Systems engineering simplified**. CRC Press, 2015.

CORONA, Erika *et al.* A review of lean-kanban approaches in the software development. **WSEAS transactions on information science and applications**, v. 10, n. 1, p. 1-13, 2013.

CZEMMEL, Jan; BAJUS, Marco; SCHÄFFER, Thomas. Identification of Criteria for the Evaluation of Reference Models—A Literature Review. **AMCIS 2024 Proceedings**, 2024.

DANKOVIC, Danijel; VRACAR, Ljubomir; PRIJIC, Aneta; PRIJIC, Zoran. An Electromechanical Approach to a Printed Circuit Board Design Course. **Ieee Transactions On Education**, [S.L.], v. 56, n. 4, p. 470-477, nov. 2013. Institute of Electrical and Electronics Engineers (IEEE), 2013.

GUÉRINEAU, Julia *et al.* Organizing the fragmented landscape of multidisciplinary product development: a mapping of approaches, processes, methods and tools from the scientific literature. **Research in Engineering Design**, v. 33, n. 3, p. 307-349, 2022.

HASSAN, Rashedul; MARKANTONAKIS, Konstantinos; AKRAM, Raja Naeem. Can you call the software in your device be firmware?. In: 2016 **IEEE 13th International Conference on e-Business Engineering (ICEBE)**. IEEE, 2016.

HAUSE, Matthew *et al.* The SysML modelling language. In: **Fifteenth European systems engineering conference**. 2006. p. 1-12.

HENDLER, Stine. Digital-physical product development: a qualitative analysis. **European Journal of Innovation Management**, v. 22, n. 2, p. 315-334, 2019.

HENDLER, Stine. Exploring coordination practices in digital–physical product development. **Journal of Manufacturing Technology Management**, v. 32, n. 3, p. 742-771, 2021.

HIRSHORN, S. R.; VOSS, L. D.; BROMLEY, L. K. NASA Systems Engineering Handbook (Report No NASA SP-2016-6105 Rev2). **National Aeronautics and Space Administration**. <https://ntrs.nasa.gov/citations/20170001761>, 2017.

IEEE STANDARDS COMMITTEE *et al.* **IEEE standard glossary of software engineering terminology**. IEEE Std, v. 610, p. 12, 1990.

INCOSE. **Systems engineering handbook: a guide for system life cycle processes and activities**. John Wiley & Sons, 2015.

INCOSE. **Systems Engineering Handbook: A" what To" Guide for All SE Practitioners**, 2004.

JUSIUK, Paulina. Influence of Pro-Environmental Attitudes on the Choice between Tangible and Virtual Product Forms. **Sustainability**, [S.L.], v. 15, n. 13, p. 10419, 1 jul. 2023. MDPI AG, 2023.

KANNENGIESSER, Udo; GERO, John S. What distinguishes a model of systems engineering from other models of designing? An ontological, data-driven analysis. **Research in Engineering Design**, v. 33, n. 2, p. 129-159, 2022.

KHAITAN, Siddhartha Kumar; MCCALLEY, James D.. Design Techniques and Applications of Cyberphysical Systems: a survey. **Ieee Systems Journal**, [S.L.], v. 9, n. 2, p. 350-365, jun. 2015. Institute of Electrical and Electronics Engineers (IEEE), 2015.

KRISHNAN, Viswanathan, ULRICH Karl. **Product Development Decisions: A Review of the Literature**. Management Science 47, (2001).

LEAU, Yu Beng *et al.* Software development life cycle AGILE vs traditional approaches. In: **International Conference on Information and Network Technology**. p. 162-167. 2012.

LOUREIRO, Guilherme Branco. **Sistematização do processo de desenvolvimento de um portador de sistemas ultrassônicos para inspeção de uniões soldadas**. 2013. 193 f. Dissertação (Mestrado) - Programa de Pós-graduação em Engenharia Mecânica, Universidade Federal de Santa Catarina, Florianópolis, 2013.

MARKLE, William H. The Manufacturing Manager's Skills. **The Manufacturing Man and His Job**, 1966.

MATHARU, Gurpreet Singh *et al.* Empirical study of agile software development methodologies: A comparative analysis. **ACM SIGSOFT Software Engineering Notes**, v. 40, n. 1, p. 1-6, 2015.

MCCARTHY, Aimee K. **Activity Mapping of Development Methods as a Decision Aid for Hardware Development Programs**. 2021. Tese de Doutorado. Monterey, CA; Naval Postgraduate School, 2021.

MCHAREK, Mehdi *et al.* Collaborative design process and product knowledge methodology for mechatronic systems. **Computers in Industry**, v. 105, p. 213-228, 2019.

MICHALIDES, Marvin *et al.* Analyzing current challenges on scaled agile development of physical products. **Procedia CIRP**, v. 119, p. 1188-1197, 2023.

MONIRUZZAMAN, A. B. M.; HOSSAIN, Dr Syed Akhter. Comparative Study on Agile software development methodologies. **Global Journal of Computer Science and Technology**, v.8, n.7, 2013.

MORGAN, James M.; LIKER, Jeffrey K. **Sistema Toyota de desenvolvimento de produto: integrando pessoas, processo e tecnologia**. São Paulo: Bookman, 2008.

MULE, Sagar et al. A new agile hybridization approach and a set of related guidelines for mechatronic product development. In: **Product Lifecycle Management Enabling Smart X: 17th IFIP WG 5.1 International Conference**, PLM 2020, Rapperswil, Switzerland, July 5–8, 2020

NASA, NPR 7123.1C **NASA Systems Engineering Processes and Requirements**, 2020.

NASA, S. E. NASA systems engineering handbook. **National Aeronautics and Space Administration, NASA/SP-2007-6105 Rev1**, 2007.

PAGAN, Rafael Perez; SILVA, Carlos Eduardo Sanches da; MELLO, Carlos Henrique Pereira. Projeto conceitual no processo de desenvolvimento de produtos eletroeletrônicos: estudos de caso em empresas incubadas. **Revista Produção Online**, [S.L.], v. 13, n. 3, p. 1089-1117, 16 ago. 2013. Associação Brasileira de Engenharia de Produção – ABEPRO, 2013.

PAHL, Gehrard *et al.* **Engineering Design: a systematic approach**. London: Springer Verlag, 2007.

PRESSMAN, Roger S.; MAXIM, Bruce R. **Engenharia de software-9 ed.** McGraw Hill Brasil, 2021.

PRODUTO. In: **MICHAELIS moderno dicionário da língua portuguesa. São Paulo: Melhoramentos**. Disponível em: <https://michaelis.uol.com.br/moderno-portugues/busca/portugues-brasileiro/produto> . Acesso em: 12 de Julho de 2023.

RAI, Preeti; DHIR, Saru. Impact of different methodologies in software development process. **International Journal of Computer Science and Information Technologies**, v. 5, n. 2, p. 1112-1116, 2014.

REINERT, Fabíola. **Integração da ergonomia no projeto de produtos com base no projeto centrado no usuário**. 2017. 181 f. Tese (Doutorado) - Curso de Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 2017.

RIGO, Cezar Antônio. **Projeto de Placas de Circuito Impresso com FPGAs para uso em ambiente espacial**. Dissertação (Mestrado) - Curso de Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Santa Catarina, Florianópolis, 2019.

ROMANO, Leonardo Nabaes. **Modelo de referência para o processo de desenvolvimento de máquinas agrícolas**. 2003. Tese (Doutorado em Engenharia Mecânica) – Programa de Pós-Graduação em Engenharia Mecânica, Universidade Federal de Santa Catarina, Florianópolis, 2003.

ROMERAL, Pedro A. de A.F.; ZANCUL, Eduardo; NAKANO, Davi. Product Development Process for complex hardware-based solutions: current trends. **Procedia Cirp**, [S.L.], v. 119, p. 885-890, 2023. Elsevier BV, 2023.

ROZENFELD, Henrique *et al.* **Gestão de desenvolvimento de produtos: uma referência para a melhoria do processo.** São Paulo, SP: Saraiva, 2006.

RUSS, Samuel. **Signal Integrity: applied electromagnetics and professional practice.** 2. ed. Cham: Springer, 2022.

SAEED, Soobia *et al.* Analysis of software development methodologies. **International Journal of Computing and Digital Systems**, v. 8, n. 5, p. 446-460, 2019.

TEJA, Ravi. **Firmware vs Software: difference between software and firmware**, 2021. Disponível em: <https://www.electronicshub.org/firmware-vs-software/>. Acesso em: 12 mar. 2024.

TOMAZELLI, Idezio João. **Uma contribuição para planejamento de produto em pequenas e médias empresas.** 2006. 209 f. Dissertação (Mestrado) - Curso de Mestrado em Engenharia Mecânica, Universidade Federal de Santa Catarina, Florianópolis, 2006.

YADAV, Ravi Shanker. Improvement in the V-Model. **International Journal of Scientific & Engineering Research**, v. 3, n. 2, p. 1-6, 2012.

APÊNDICE A – PESQUISA EXPLORATÓRIA

De forma complementar ao referencial teórico apresentado, foram realizadas, durante o ano de 2023, 18 entrevistas semiestruturadas com pesquisadores de uma empresa de desenvolvimento de sistemas embarcados, a fim de entender sua percepção do desenvolvimento de produtos multidisciplinares e, também, identificar os processos de desenvolvimento relacionados às suas áreas de conhecimento. Tal empresa possui diversos projetos que se enquadram no conceito de produtos multidisciplinares apresentado. Há uma divisão interna de equipes técnicas, a qual enquadra seus pesquisadores, de distintas formações, em áreas de conhecimento, dentre as quais:

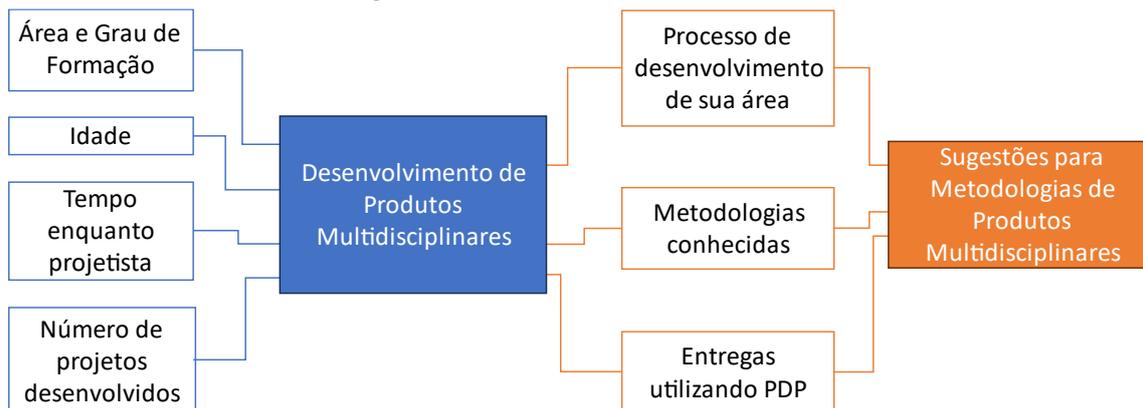
- Inteligência artificial;
- Software PC/Mobile;
- Mecânica;
- Controle, Automação e Instrumentação;
- Software Embarcado (Firmware);
- Hardware.

Conforme o escopo desta dissertação, as 6 áreas listadas podem ser enquadradas em 3 DEs, conforme os critérios apresentados na seção anterior:

- Desenvolvimento mecânico: com a equipe de Mecânica propriamente dita;
- Desenvolvimento eletrônico: abrange as equipes de Hardware e Controle, Automação e Instrumentação;
- Desenvolvimento de software: abrange as equipes de Inteligência artificial, Software PC/Mobile e Software Embarcado;

Cada entrevista durou cerca de 90 minutos e foi conduzida pelo autor, com apoio de um questionário semiestruturado. A Figura 43 apresenta os pontos tratados no questionário. O questionário completo está contido no Apêndice B.

Figura 43 - Conteúdo das entrevistas.

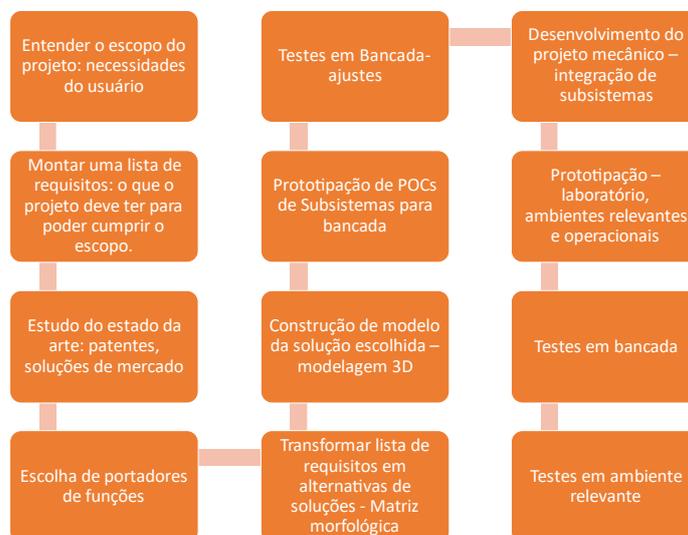


Fonte: o autor.

A média de idade dos entrevistados é de 33,1 anos, e de experiência enquanto desenvolvedores é 10,3 anos. Nesse tempo de experiência constatou-se uma média de 12,5 projetos desenvolvidos, o que resultaria em, aproximadamente, 1,2 projetos por ano. Durante as entrevistas foram identificadas 11 áreas de conhecimento distintas e 3 níveis de formação, sendo 10 Mestres, 5 Doutores e 3 Graduados.

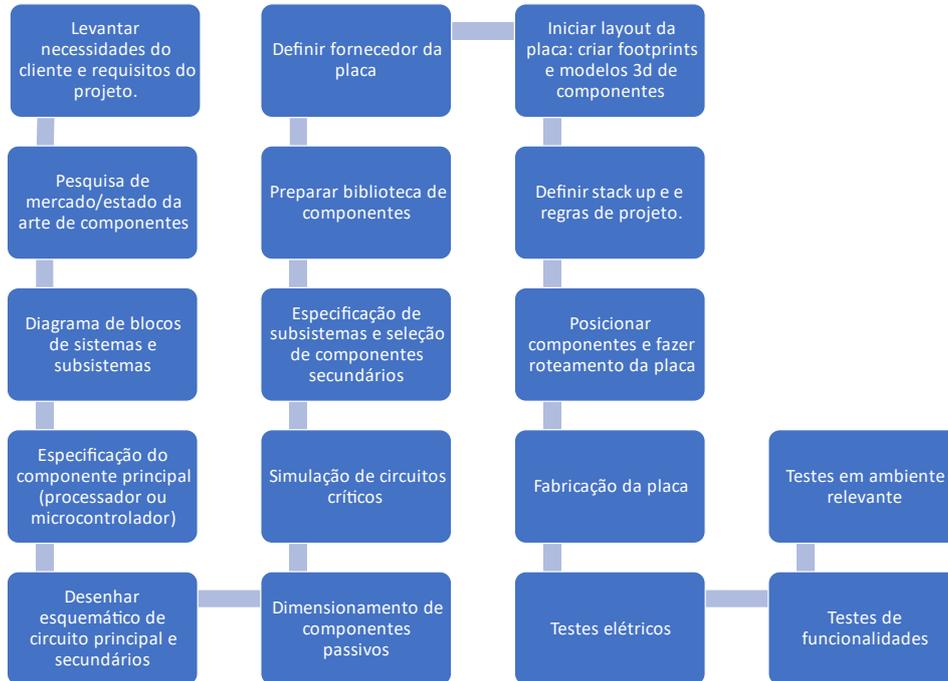
A Figura 44, a Figura 45 e a Figura 46 apresentam um resumo construído de acordo com as informações citadas, de maneira espontânea pelos entrevistados, sobre os processos usados corriqueiramente, pelos mesmos, para o desenvolvimento mecânico, eletrônico e de software respectivamente.

Figura 44 - Processo de desenvolvimento dos pesquisadores de desenvolvimento mecânico.



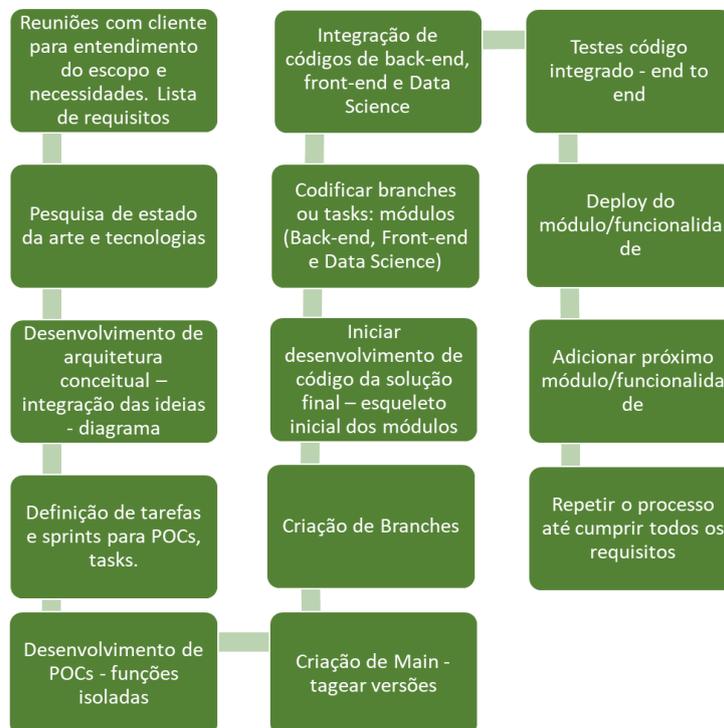
Fonte: o autor.

Figura 45 - Processo de desenvolvimento dos pesquisadores de desenvolvimento eletrônico.



Fonte: o autor.

Figura 46 - Processo de desenvolvimento dos pesquisadores de desenvolvimento de software.



Fonte: o autor.

Além dos processos utilizados, foi avaliado o conhecimento dos entrevistados em relação a modelos formais de desenvolvimento. A Tabela 4 reúne os modelos mencionados. Nota-se nessas menções a ocorrência frequente de modelos direcionados para software e modelos de PDP (especialmente pela equipe de pesquisadores de mecânica), havendo pouca menção a modelos de suporte ao desenvolvimento eletrônico.

Tabela 4 - Modelos mencionados.

Modelos	Tipo	Ocorrências
Modelos ágeis (genérico)	Software	5
Cascata	Software	3
DevOps	Software	3
PRODIP	PDP	3
Pahl e Beitz	PDP	2
SCRUM	Software	2
BDD	Software	1
DFM	PDP/Eletrônica	1
Engenharia de Sistemas	Sistemas	1
Engenharia reversa	PDP	1
Kanban	Software	1
Lean PDP	PDP	1
MDO – Multidisciplinary Design and Optimization	Sistemas	1
MLOPS	Software	1
Modelo de referência (Rozenfeld)	PDP	1
Modelo em V	Software	1
RUP	Software	1
TTD	Software	1
XP	Software	1

Fonte: o autor.

Outro ponto abordado durante a pesquisa foi o entendimento de entregáveis das distintas áreas de conhecimento de acordo com um mesmo modelo. Para esse entendimento foi utilizado um modelo baseado em Pahl *et al.* (2007), por ser de fácil entendimento e difundido como referência para o PDP. A Figura 3 foi apresentada como referência para os entrevistados. O Quadro 18, o Quadro 19 e o Quadro 20 mostram, respectivamente, os entregáveis relacionados ao desenvolvimento do domínio da Mecânica, Eletrônica e Software, de acordo com o processo do PDP.

Quadro 18 - Entregáveis da equipe de Mecânica.

Fase	Entregáveis
Esclarecimento da Tarefa	Escopo e premissas do projeto; Lista de requisitos;
Projeto Conceitual	Lista de funções para o projeto; Lista de soluções para cada função; Formulação de conceitos.
Projeto Preliminar	Descrição e testes de conceitos desenvolvidos; Modelo 3D de conceitos; Desenhos técnicos de peças e protótipos; Bill of Materials (BoMs).
Projeto Detalhado	Procedimentos de Fabricação; Procedimentos de montagem; Manual de operação;

Fonte: o autor.

Quadro 19 - Entregáveis da equipe de Eletrônica.

Fase	Entregáveis
Esclarecimento da Tarefa	Lista de requisitos;
Projeto Conceitual	Proposição de arquiteturas para atender aos requisitos; Lista de opções de componentes; Escolha de componentes principais.
Projeto Preliminar	Esquemático da PCI; <i>Layout</i> da placa; Testes em bancada e ambiente relevante; Arquivos fonte e pacote de fabricação da placa (<i>Gerbers, BoM, Pick and place</i>).
Projeto Detalhado	Estudo de componentes para lote piloto; Reprojeto de placa; Testes de homologação e compatibilidade; Otimização de arquivos do ECAD (projeto, Gerber, BoM, Stack up);

Fonte: o autor.

Quadro 20 - Entregáveis da equipe de Software.

Fase	Entregáveis
Esclarecimento da Tarefa	Lista de demandas do cliente; Lista de requisitos.
Projeto Conceitual	Definição de arquitetura do projeto; Diagrama de fluxo de dados.
Projeto Preliminar	Definição de tecnologias; Apresentação de resultados: documentos, prints, vídeos; manual de uso e repasse para fornecedores.
Projeto Detalhado	Versão testada, validada e estável; Escalabilidade (para versões em nuvem); Adequar para escalabilidade (monitores e celulares); Revisar implementação de camada de segurança; Integração com ferramentas comerciais.

Fonte: o autor.

Ao comparar os entregáveis dos quadros com o modelo de Pahl *et al.* (2007), pode-se inferir que:

- Os entregáveis elicitados pela equipe de mecânica no Quadro 18 aderem à sequência: lista de requisitos, conceito, *layout* de forma, *layout* aprimorado, documentos de produção sugerida pela Figura 3;
- Os entregáveis citados pela equipe de eletrônica seguem uma sequência: requisitos, escolha de componentes, esquemático, fabricação e reprojeto, a qual, apesar de certa similaridade, distingue da Figura 3;
- Os entregáveis descritos pela equipe de software por sua vez diferem bastante do modelo de Pahl e Beitz após a fase de projeto conceitual.

Essa comparação reforça que os modelos de PDP têm uma maior especificidade para o desenvolvimento de produtos mecânicos. Mesmo um modelo bastante difundido, como o de Pahl *et al.* (2007), traz em sua própria linguagem termos mais utilizados no desenvolvimento mecânico (escolha de materiais, *layout* de forma, documentos de montagem), o que pode causar certa dificuldade em utilizá-lo em distintas áreas do conhecimento.

Por fim, foi solicitado aos entrevistados, de acordo com sua experiência, sugestões e necessidades para um modelo de referência para produtos

multidisciplinares. A Tabela 5 apresenta os principais pontos citados pelos entrevistados.

Tabela 5 - Sugestões para Modelos Multidisciplinares.

Sugestões	Ocorrências
Definir pontos de integração entre áreas	7
Definir processos específicos para cada área	7
Definir entregáveis de cada área por ciclo de vida	5
Ter pessoas como referência de cada área no projeto	4
Definir um modelo para a empresa	4
Difundir um modelo através de treinamento	2
Responsável técnico deve ter uma visão multidisciplinar	2
Aprofundar análise de custo	1
Atualizar requisitos ao longo do projeto	1
Ciclos iterativos com momentos de pausa e análise	1
Estabelecer cronogramas de compras	1
Definir granularidade de cronograma	1
Utilizar Engenharia simultânea	1
Melhorar documentação	1
Padronizar nomes de variáveis	1
Utilizar abordagem funcional ao invés de estrutural	1
Utilizar modelos ágeis	1

Fonte: o autor.

**APÊNDICE B – QUESTIONÁRIO: PDP APLICADO A PROJETOS
MULTIDISCIPLINARES**

Data: __/__/__

Nome:

Idade:

Formação:

Área de conhecimento na empresa:

Tempo como desenvolvedor:

Número de projetos desenvolvidos (aproximadamente):

1. Descreva resumidamente sua área de conhecimento:

2. Descreva o processo de desenvolvimento de produtos sua área na empresa.

3. Quais os modelos/ /processos/frameworks de desenvolvimento de produtos você conhece?

4. De acordo com a Figura 3²⁶, o que você entende por entregas da sua área em cada fase?
 - a) Projeto Informacional

 - b) Projeto Conceitual

 - c) Projeto Preliminar

²⁶ Essa figura se refere ao modelo de Pahl e Beitz, apresentado no capítulo 2.

d) Projeto Detalhado

5. Você considera que alguma das metodologias acima é ideal para projetos multidisciplinares? Por quê?

6. Quais suas sugestões para melhoria de projetos multidisciplinares?

Empresa:	Nome da empresa		
Projeto:	Nome do projeto		
Revisão:	RAP – Revisão de Arquitetura do Produto		
DE:	Nome da DE	Data:	dd/mm/aaaa

Líder da DE: Nome do líder da DE
Equipe de DE: Integrante 1
Integrante 2
Integrante 3
Engenheiro de Sistemas: Nome do engenheiro de sistemas

Apresentação do Conteúdo

Modelagem diagramática de funções e elementos do produto.

Critério de sucesso: Demonstração de funções e elementos do produto, incluindo interfaces internas e externas com outros produtos.

Revisão aprovada:

Sim

Não

Recomendações para Revisão integrada	Ações necessárias
Recomendação 1	Ação 1
Recomendação 2	Ação 2
Recomendação 3	Ação 3

Líder de DE

Engenheiro de Sistemas

Empresa:	Nome da empresa		
Projeto:	Nome do projeto		
Revisão:	RCP – Revisão de Conceito do Produto		
DE:	Nome da DE	Data:	dd/mm/aaaa

Líder da DE: Nome do líder da DE
Equipe de DE: Integrante 1
Integrante 2
Integrante 3
Engenheiro de Sistemas: Nome do engenheiro de sistemas

Apresentação do Conteúdo

Registro visual de conceito ou protótipos iniciais.

Critério de sucesso: Os conceitos devem se mostrar viáveis para seguir o desenvolvimento.

Revisão aprovada:

Sim

Não

Recomendações para Revisão integrada	Ações necessárias
Recomendação 1	Ação 1
Recomendação 2	Ação 2
Recomendação 3	Ação 3

Líder de DE

Engenheiro de Sistemas

Empresa:	Nome da empresa		
Projeto:	Nome do projeto		
Revisão:	RTP – Revisão de Testes do Produto		
DE:	Nome da DE	Data:	dd/mm/aaaa

Líder da DE: Nome do líder da DE
Equipe de DE: Integrante 1
Integrante 2
Integrante 3
Engenheiro de Sistemas: Nome do engenheiro de sistemas

Apresentação do Conteúdo

Registro visual e relatórios de testes.

Critério de sucesso: Testes dos produtos devem se mostrar bem-sucedidos, os requisitos dos produtos devem se mostrar minimamente atendidos.

Revisão aprovada:

Sim

Não

Recomendações para Revisão integrada	Ações necessárias
Recomendação 1	Ação 1
Recomendação 2	Ação 2
Recomendação 3	Ação 3

Líder de DE

Engenheiro de Sistemas

APÊNDICE D – REVISÕES INTEGRADAS

Empresa:	Nome da empresa
Projeto:	Nome do projeto
Revisão:	RIO – Revisão Integrada de Objetivos
Data:	dd/mm/aaaa

Engenheiro de Sistemas: Nome do líder da DE
Líderes de DE: Integrante 1
Integrante 2
Integrante 3
Cliente: Nome do engenheiro de sistemas

Apresentação do Conteúdo

Lista contendo as necessidades e objetivos dos *Stakeholders* e proposta inicial de conceitos para atingir esses objetivos, ou seja, qual a missão do produto multidisciplinar.

Critério de sucesso: Necessidades dos *Stakeholders* devem ser esclarecidas e documentadas e a proposta deve demonstrar que pode as solucionar.

Revisão aprovada:

Sim

Não

Recomendações	Ações necessárias
Recomendação 1	Ação 1
Recomendação 2	Ação 2
Recomendação 3	Ação 3

Engenheiro de Sistemas

Cliente

Empresa:	Nome da empresa
Projeto:	Nome do projeto
Revisão:	RIR – Revisão Integrada de Requisitos
Data:	dd/mm/aaaa

Engenheiro de Sistemas: Nome do líder da DE
Líderes de DE: Integrante 1
Integrante 2
Integrante 3
Cliente: Nome do engenheiro de sistemas

Apresentação do Conteúdo

Lista contendo descrição, razão e método de verificação de requisitos funcionais e de desempenho para o produto multidisciplinar e suas interfaces.

Critério de sucesso: Os requisitos do produto multidisciplinar devem estar claros, concisos e verificáveis.

Revisão aprovada:

Sim

Não

Recomendações	Ações necessárias
Recomendação 1	Ação 1
Recomendação 2	Ação 2
Recomendação 3	Ação 3

Engenheiro de Sistemas

Cliente

Empresa:	Nome da empresa
Projeto:	Nome do projeto
Revisão:	RIA – Revisão Integrada de Arquitetura
Data:	dd/mm/aaaa

Engenheiro de Sistemas: Nome do líder da DE
Líderes de DE: Integrante 1
Integrante 2
Integrante 3
Cliente: Nome do engenheiro de sistemas

Apresentação do Conteúdo

Modelagem de arquitetura do produto multidisciplinar, contendo a modelagem dos produtos das DEs, interfaces internas e externas.

Critério de sucesso: A arquitetura proposta deve estar de acordo com os requisitos, os quais devem estar alocados para todos os elementos funcionais do produto multidisciplinar.

Revisão aprovada:

Sim

Não

Recomendações	Ações necessárias
Recomendação 1	Ação 1
Recomendação 2	Ação 2
Recomendação 3	Ação 3

Engenheiro de Sistemas

Cliente

Empresa:	Nome da empresa
Projeto:	Nome do projeto
Revisão:	RIPC – Revisão Integrada Preliminar de Conceitos
Data:	dd/mm/aaaa

Engenheiro de Sistemas: Nome do líder da DE
Líderes de DE: Integrante 1
Integrante 2
Integrante 3
Cliente: Nome do engenheiro de sistemas

Apresentação do Conteúdo

Apresentação de forma escrita ou gráfica do conceito multidisciplinar do produto.

Critério de sucesso: O conceito proposto deve demonstrar que cumpre todos os requisitos de sistema dentro de um risco, custo e cronograma aceitáveis e estabelece a base para evolução do mesmo.

Revisão aprovada:

Sim

Não

Recomendações	Ações necessárias
Recomendação 1	Ação 1
Recomendação 2	Ação 2
Recomendação 3	Ação 3

Engenheiro de Sistemas

Cliente

Empresa:	Nome da empresa
Projeto:	Nome do projeto
Revisão:	RICC – Revisão Integrada Crítica de Conceitos
Data:	dd/mm/aaaa

Engenheiro de Sistemas: Nome do líder da DE
Líderes de DE: Integrante 1
Integrante 2
Integrante 3
Cliente: Nome do engenheiro de sistemas

Apresentação do Conteúdo

Apresentação da evolução dos conceitos mecânico, eletrônico, de software e interfaces internas e externas.

Critério de sucesso: A documentação deve demonstrar que a maturidade dos produtos é adequada e integrada para prosseguir com a realização dos mesmos.

Revisão aprovada:

Sim

Não

Recomendações	Ações necessárias
Recomendação 1	Ação 1
Recomendação 2	Ação 2
Recomendação 3	Ação 3

Engenheiro de Sistemas

Cliente

Empresa:	Nome da empresa
Projeto:	Nome do projeto
Revisão:	RIRP – Revisão Integrada de Realização de Produtos
Data:	dd/mm/aaaa

Engenheiro de Sistemas: Nome do líder da DE
Líderes de DE: Integrante 1
Integrante 2
Integrante 3
Cliente: Nome do engenheiro de sistemas

Apresentação do Conteúdo

Checklist de produção e integração de produtos, interfaces, produtos de apoio e mão de obra.

Critério de sucesso: O checklist deve assegurar que todos os produtos, seus segmentos, componentes e produtos de apoio para integração no sistema, assim como infraestrutura de integração, mão de obra de suporte e planos e procedimentos estão disponíveis.

Revisão aprovada:

Sim

Não

Recomendações	Ações necessárias
Recomendação 1	Ação 1
Recomendação 2	Ação 2
Recomendação 3	Ação 3

Engenheiro de Sistemas

Cliente

Empresa:	Nome da empresa
Projeto:	Nome do projeto
Revisão:	RIPM – Revisão Integrada do Produto Multidisciplinar
Data:	dd/mm/aaaa

Engenheiro de Sistemas: Nome do líder da DE

Líderes de DE: Integrante 1
Integrante 2
Integrante 3

Cliente: Nome do engenheiro de sistemas

Apresentação do Conteúdo

Relatório de integração e testes preliminares do produto multidisciplinar.

Critério de sucesso: Os produtos realizados devem se integrar de acordo com a arquitetura proposta.

Revisão aprovada:

Sim

Não

Recomendações	Ações necessárias
Recomendação 1	Ação 1
Recomendação 2	Ação 2
Recomendação 3	Ação 3

Engenheiro de Sistemas

Cliente

Empresa:	Nome da empresa
Projeto:	Nome do projeto
Revisão:	RVPM – Revisão de Verificação do Produto Multidisciplinar
Data:	dd/mm/aaaa

Engenheiro de Sistemas: Nome do líder da DE

Líderes de DE: Integrante 1
Integrante 2
Integrante 3

Cliente: Nome do engenheiro de sistemas

Apresentação do Conteúdo

Relatório de integração e testes preliminares do produto multidisciplinar.

Critério de sucesso: Os produtos realizados devem se integrar de acordo com a arquitetura proposta.

Revisão aprovada:

Sim

Não

Recomendações	Ações necessárias
Recomendação 1	Ação 1
Recomendação 2	Ação 2
Recomendação 3	Ação 3

Engenheiro de Sistemas

Cliente

Empresa:	Nome da empresa
Projeto:	Nome do projeto
Revisão:	RAPM – Revisão de Aceitação do Produto Multidisciplinar
Data:	dd/mm/aaaa

Engenheiro de Sistemas: Nome do líder da DE

Líderes de DE: Integrante 1
Integrante 2
Integrante 3

Cliente: Nome do engenheiro de sistemas

Apresentação do Conteúdo

Relatório de validação e Aceitação do cliente.

Critério de sucesso: O produto multidisciplinar a deve cumprir as necessidades expressas pelos stakeholders. Pode se considerar pronto para o lançamento.

Revisão aprovada:

Sim

Não

Recomendações	Ações necessárias
Recomendação 1	Ação 1
Recomendação 2	Ação 2
Recomendação 3	Ação 3

Engenheiro de Sistemas

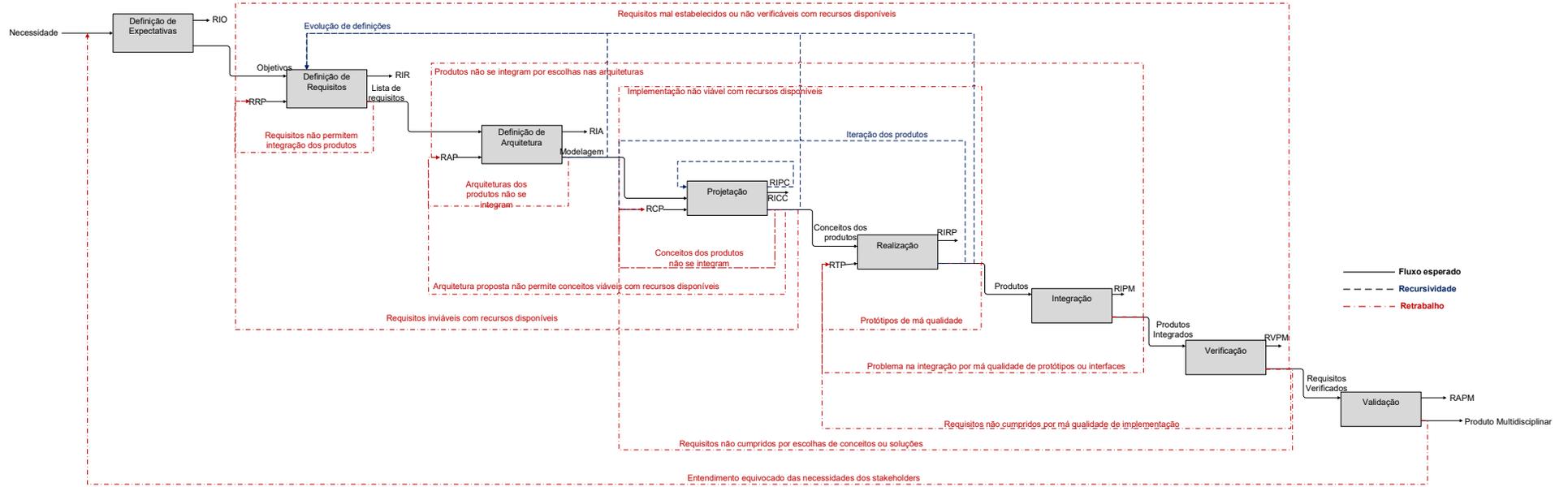
Cliente

APÊNDICE E – FLUXOS DO PROCESSO

O processo do MDPM tem um fluxo esperado, no qual, a partir de uma necessidade, gera-se uma lista de requisitos, seguida de uma modelagem integrada, então conceitos dos produtos, a seguir os produtos construídos, integrados, o produto integrado é verificado, validado e, enfim, o produto multidisciplinar desenvolvido. Tais fases não são completamente lineares: produtos mecânicos, eletrônicos e, especialmente, softwares são muitas vezes desenvolvidos conforme versões, ou seja, de forma incremental; algumas particularizações de requisitos são possíveis somente quando o projeto evolui em suas fases, aumentando o conhecimento da tecnologia. Há uma iteratividade prevista especialmente entre as fases de Requisitos, Arquitetura, Projeção e Realização e Implementação.

Ainda, o desenvolvimento e a inovação, especialmente tratando de desenvolvimentos paralelos, são passíveis de incertezas, as quais podem desviar do fluxo esperado, causando retrabalho. A **Erro! Fonte de referência não encontrada.** ilustra os fluxos de iteração e os retrabalhos que cada desvio de fluxo pode acarretar.

Figura 47 - Fluxos do processo.



Fonte: o autor.

Nota-se em azul os fluxos de recursividade e iteração, e em vermelho estão destacados os possíveis retrabalhos, que acarretam revisão de decisões de projeto.

Dentre esses fluxos possíveis nas fases, pode-se citar:

- Na fase de Definição de Requisitos há uma recursividade utilizada como controle de alguns requisitos que podem ser melhor estabelecidos quando há evoluções na definição arquitetura, projeção ou realização. Nessa fase ainda pode ocorrer retrabalho devido a problemas na integração dos requisitos dos diferentes produtos;
- Na fase de Definição de Arquitetura também pode ocorrer retrabalho em arquiteturas de produtos que não são integráveis entre si;
- Na fase de Projeção há uma revisão integrada durante a fase, a qual, de maneira recursiva, realimenta a evolução dos conceitos dos produtos. Tais conceitos novamente podem passar por retrabalho caso se note que os requisitos estabelecidos ou as decisões de arquitetura não permitam desenvolvimento de conceitos viáveis de acordo com recursos (principalmente humano, técnico, custo, prazo ou tecnologia) disponíveis. E, novamente, os conceitos de cada produto devem se integrar uns aos outros para evitar retrabalho;
- Na fase de realização e implementação pode haver retrabalho devido à qualidade dos protótipos construídos ou, até mesmo, relacionados à escolha de conceitos não realizáveis de acordo com os recursos disponíveis.
- Na fase de integração, uma vez que os protótipos estejam construídos, pode também haver retrabalho ou revisão de projeto ocasionada por problemas em suas interfaces, geradas por escolhas inadequadas na arquitetura ou na implementação.
- Na fase de Verificação, com o sistema integrado, há um momento crucial: o sistema ou os produtos podem não cumprir requisitos por qualidade na implementação, na escolha de soluções, ou até mesmo requisitos mal estabelecidos;
- Na fase de validação, por fim, um produto ou sistema pode passar por todas as etapas de desenvolvimento, integração e testes e não cumprir o que o *Stakeholder* esperava devido a um entendimento equivocado de suas necessidades.

Os fluxos ilustrados na Figura 47 mostram que quanto mais tardio no processo se encontra o desvio, maior o caminho a ser percorrido devido a retrabalhos. As fases, mecanismos, especialmente, as revisões propostas no modelo buscam um fluxo que identifique as incoerências e evite o retrabalho. Mesmo em momentos que as revisões são aprovadas e o desenvolvimento segue um fluxo esperado, conferência e até mesmo pontuais correções em fases passadas (em especial na definição de requisitos) podem e devem ocorrer.

ANEXO A – BOAS PRÁTICAS NO PROJETO DE PCIS

Apesar de não ter sido encontrado na literatura um modelo para desenvolvimento eletrônico, constam algumas recomendações para projeto de PCIs. Conforme Rigo (2019) e Russ (2022), são boas práticas:

- Realizar um processo iterativo entre esquemático e seleção de componentes: raramente todos os componentes podem ser selecionados antes do início do esquemático e, frequentemente, informações trazidas pelo esquemático devem causar uma revisão na seleção dos componentes;
- Usar ferramentas de apoio para se gerar a lista de componentes: em projetos com tempo de execução curto, durante o desenvolvimento do esquemático, há ferramentas que permitem ao projetista exportar uma lista de peças, ou seja, uma primeira versão da BoM (*Bill of Materials*). Essa lista pode ser utilizada para iniciar aquisição de componentes enquanto a placa passa para os próximos processos de desenvolvimento;
- Revisar por pares o esquemático uma vez que esteja completo;
- Escolher o fabricante antes de definir o *layout* da placa, devido a diferentes métodos e máquinas na confecção das placas;
- Envolver os fornecedores desde as fases iniciais, pois o conhecimento da empresa e fornecimento de amostra pode auxiliar decisões de projeto;
- Considerar margens de tolerância, padrões de fabricação do fornecedor, materiais disponíveis e as mínimas dimensões implementáveis para todas as especificações do projeto. Podem ser necessárias iterações com o fabricante;
- Considerar o custo da fabricação, necessidades de roteamento, interferência eletromagnética, armazenamento de energia, dissipação de calor e higroscopia ao se definir o número de camadas;
- Considerar simetria, especificações do fabricante, dissipação de calor, higroscopia, impedância, níveis de tensão e corrente e interferência eletromagnética para definir espessura das trilhas, camadas de cobre e material isolante;
- Verificar se trilhas e vias atendem às mínimas distâncias de isolamento e comprimento, níveis de tensão, corrente, impedância e ruído;
- Definir a localização dos componentes de acordo com a geometria, dimensões e montagem dos componentes;

- Realizar revisão por pares o *layout* (assim como no esquemático): como os sinais são roteados, onde os planos são, onde os componentes estão, tamanho e forma da placa para evitar problemas de integridade de sinal.

ANEXO B – VALORES DA ABORDAGEM ÁGIL

Quadro 21 - Valores da abordagem ágil.

1. Indivíduos e interações acima de processos e ferramentas.
2. Software funcional acima de documentação intensiva.
3. Colaboração com o cliente acima de negociação contratual.
4. Resposta às mudanças acima de seguir um planejamento.

Fonte: adaptado de Alsaqqa, Sawalha e Abdel-nabi (2020).

ANEXO C – PRINCÍPIOS DA ABORDAGEM ÁGIL

Quadro 22 - Princípios da abordagem ágil.

Princípio 1: A maior prioridade é a satisfação do cliente através de entrega antecipada e contínua de software de valor.

Princípio 2: Mudanças em requisitos são bem-vindas, mesmo durante desenvolvimento adiantado.

Princípio 3: Entrega de software frequente, de poucas semanas a poucos meses, preferencialmente na menor escala de tempo.

Princípio 4: Equipes de negócio e desenvolvimento devem trabalhar em conjunto diariamente durante o projeto.

Princípio 5: Construir projetos com indivíduos motivados. Fornecer ambiente e apoio necessários e confiar a eles a entrega do trabalho.

Princípio 6: O método mais eficiente e efetivo de transmitir informações em uma equipe de desenvolvimento é pela comunicação pessoal.

Princípio 7: Software funcional é a medida primária de progresso.

Princípio 8: Processos ágeis promovem ambiente sustentável.

Princípio 9: Atenção contínua à excelência técnica e projeto de qualidade traz maior agilidade.

Princípio 10: Simplicidade: a arte de maximizar a quantidade de trabalho não realizado é essencial.

Princípio 11: As melhores arquiteturas, requisitos e projetos emergem de equipes auto-organizadas.

Princípio 12: Em intervalos regulares, a equipe deve refletir em como se tornar mais efetiva e ajustar seu comportamento de acordo.

Fonte: adaptado de Alsaqqa, Sawalha e Abdel-nabi (2020).

ANEXO D – REVISÕES DA ENGENHARIA DE SISTEMAS

Quadro 23 - Revisões da Engenharia de Sistemas²⁷.

MCR (<i>Mission Concept Review</i>) – Revisão de Conceito da Missão	
Descrição:	Declama as necessidades de missão e avalia os objetivos propostos e conceitos para atingir esses objetivos.
Ocorrência:	Deve ser entregue antes de iniciar a fase de desenvolvimento de conceito (Fase A).
Critério de sucesso:	Demonstrar que a proposta de missão cumpre as necessidades do cliente e tem qualidade o suficiente para avançar seus estudos na Fase A.
SRR (<i>System Requirements Review</i>) – Revisão de Requisitos do Sistema	
Descrição:	Avalia os requisitos funcionais e de desempenho definidos para o sistema, plano preliminar de projeto e a segurança que os requisitos e conceito selecionados satisfazem a missão.
Ocorrência:	Durante a Fase de Desenvolvimento de Conceito (Fase A) e antes de conduzir a SDR ou MDR.
Critério de sucesso:	Estabelecer os requisitos do projeto e conduzir a uma decisão formal para prosseguir com a implementação do projeto
MDR (<i>Mission Definition Review</i>) /SDR (<i>System Definition Review</i>) – Revisão de Definição da Missão/Sistema	
Descrição:	Os títulos se distinguem por aplicação da NASA entre missões tripuladas (SDR) ou robóticas (MDR). Avalia se a arquitetura proposta está de acordo com os requisitos funcionais e de desempenho e se os requisitos foram alocados para todos os elementos funcionais do sistema.
Ocorrência:	Durante a Fase de Desenvolvimento de Conceito e antes do início do Projeto Preliminar.
Critério de sucesso:	Suportar a decisão de prosseguir o desenvolvimento arquitetura e projeção do sistema e a tecnologia necessária e/ou prover uma base para a estratégia de aquisições do sistema.
PDR (<i>Preliminary Design Review</i>) - Revisão Preliminar de Projeto	
Descrição:	Demonstra que o projeto preliminar atinge todos os requisitos de sistema dentro de um risco, custo e cronograma aceitáveis e estabelece a base para prosseguimento no projeto detalhado. Demonstra que as opções corretas de projeto foram selecionadas, interfaces identificadas e métodos de validação descritos.
Ocorrência:	Próximo à finalização da Fase de Projeto Preliminar

²⁷ Algumas revisões, como a PRR (*Production Readiness Review* ou Revisão de Prontidão da Produção), FRR (*Flight Readiness Review* ou Revisão de prontidão para voo) e PFAR (*Post-Flight Assessment Review* ou Revisão de Avaliação pós-voo) não estão descritas por serem de alta especificidade para missões espaciais e, portanto, de pouca aplicabilidade a esse estudo.

como a última revisão na Fase de Formulação.
Critério de sucesso: Aprovar um <i>Baseline</i> de Projeto e obter autorização para prosseguir à Fase de Implementação e Projeto Final.
CDR (<i>Critical Design Review</i>) – Revisão Crítica de Projeto
Descrição: Demonstra que a maturidade do projeto é adequada para prosseguir com fabricação, montagem, integração e teste. Determina se o esforço técnico está apropriado para completar o desenvolvimento do sistema e atingir requisitos de desempenho da missão dentro do custo e cronograma previstos.
Ocorrência: Durante a Fase de Projeto Final (Fase C).
Critério de sucesso: Finalizar a <i>baseline</i> de construção e aprovar os planos de produção e verificação. Autorizar a codificação de software entregável de acordo com a <i>baseline</i> de construção e testes de qualificação e integração.
SIR (<i>System Integration Review</i>) - Revisão para a integração do Sistema
Descrição: Assegura que os segmentos, componentes e subsistemas para integração no sistema, assim como infraestrutura de integração, mão de obra de suporte e planos e procedimentos de integração estão dentro do cronograma.
Ocorrência: Ao final da Fase de Projeto Final e antes do início da Fase de Montagem, Integração e Testes.
Critério de sucesso: Aprovar a <i>Baseline</i> “As-Built”, Planos de verificação. Autorizar a liberação dos desenhos de integração.
SAR (<i>System Acceptance Review</i>) - Revisão de Aceitação do Sistema
Descrição: Verifica a completude de produtos específicos em relação a seu nível de maturidade esperado e avalia conformidade com as expectativas de cliente. Assegura que o sistema tem maturidade técnica para autorizar envio ao local de operação.
Critério de sucesso: Sistema aceito pelo cliente e autorizar o envio para local de operação e instalar o software para uso operacional.
ORR (<i>Operational Readiness Review</i>) – Revisão de Prontidão Operacional
Descrição: Examina as características reais do sistema e os procedimentos utilizados na operação. Garante que todos os produtos de hardware e software do sistema e de suporte, assim como documentação de usuário refletem o sistema entregue.
Critério de sucesso: Sistema pronto para iniciar operação normal.
PLAR (<i>Post-Launch Assessment Review</i>) – Revisão de avaliação pós-lançamento
Descrição: Avalia a prontidão do sistema para seguir com as operações. Avalia o status, desempenho e capacidades demonstradas do projeto desde o lançamento. Avalia a prontidão da mudança de responsabilidade da equipe de desenvolvimento para a organização de operações.
Ocorrência: Logo após o lançamento.
Critério de sucesso: Sistema pronto para assumir operações de rotina.
CERR (<i>Critical Event Readiness Review</i>) – Revisão de Prontidão para Evento Crítico

Descrição: Confirma a prontidão para execução de atividades críticas na operação.
Critério de sucesso: Sistema pronto para assumir operações de rotina.
DR (<i>Decommissioning Review</i>) – Revisão de Descomissionamento
Descrição: Confirma a decisão de descomissionar o sistema e avalia a prontidão para o descomissionamento seguro do sistema e seus itens.
Ocorrência: Normalmente perto do final da operação após a realização dos objetivos planejados ou devido a algum evento que leve ao término prematuro da missão.
Critério de sucesso: Garantir que o processo descomissionamento do sistema é apropriado e efetivo.
DRR (<i>Disposal Readiness Review</i>) – Revisão de Prontidão de Descarte
Descrição: Confirma a prontidão para descarte dos itens do sistema.
Ocorrência: Quando os itens do sistema estiverem prontos para descarte.
Critério de sucesso: Garantir que o processo de descarte dos itens do sistema é apropriado e efetivo.

Fonte: adaptado de Hirshorn, Voss & Bromley (2017)