



UNIVERSIDADE
ESTADUAL DE LONDRINA

JOÃO MAURÍCIO HYPÓLITO

GAIA-SANM:
FRAMEWORK PARA GERENCIAMENTO DE PROJETOS DE
PROCESSOS DE SOFTWARE

JOÃO MAURÍCIO HYPÓLITO

GAIA-SANM:
FRAMEWORK PARA GERENCIAMENTO DE PROJETOS DE
PROCESSOS DE SOFTWARE

Dissertação apresentada ao Programa de Mestrado em Ciência da Computação da Universidade Estadual de Londrina para obtenção do título de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Rodolfo Miranda de Barros.

Coorientador: Prof. Dr. Vitor Valério de Souza Campos.

Londrina
2017

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UEL

Hypólito, João Maurício.

GAIA-SANM: Framework para Gerenciamento de Projetos de Processos de Software / João Maurício Hypólito. - Londrina, 2017.
69f. : il.

Orientador: Prof. Dr. Rodolfo Miranda de Barros.

Coorientador: Prof. Dr. Vitor Valério de Souza Campos.

Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual de Londrina, Centro de Ciências Exatas, Programa de Pós-Graduação em Ciência da Computação, 2017.

Inclui bibliografia.

1. Gerenciamento de Projeto - Tese. 2. Projeto de Software - Tese. 3. Nível de Maturidade - Tese. I. Barros, Prof. Dr. Rodolfo Miranda de . II. Campos, Prof. Dr. Vitor Valério de Souza . III. Universidade Estadual de Londrina. Centro de Ciências Exatas. Programa de Pós-Graduação em Ciência da Computação. IV. Título.]

JOÃO MAURÍCIO HYPÓLITO

GAIA-SANM:
FRAMEWORK PARA GERENCIAMENTO DE PROJETOS DE
PROCESSOS DE SOFTWARE

Dissertação apresentada ao Programa de Mestrado em Ciência da Computação da Universidade Estadual de Londrina para obtenção do título de Mestre em Ciência da Computação.

BANCA EXAMINADORA

Orientador: Prof. Dr. Rodolfo Miranda de Barros
Universidade Estadual de Londrina - UEL

Prof. Dr. Jacques Duílio Brancher
Universidade Estadual de Londrina - UEL

Prof. Dr. Lisandro Rogério Modesto
Faculdade de Apucarana - FAP

Londrina, 6 de setembro de 2017.

*Este trabalho é dedicado ao Arquiteto do Universo, por existir em tudo,
à minha família, pela paciência,
e aos meus pais, por minha existência.*

AGRADECIMENTOS

Às pessoas que tiveram a calma de ouvir por mais de uma vez o que se apresenta neste trabalho. Estas pessoas tiveram '*insights*' e pouco a pouco foram ajudando a montar o mosaico do trabalho. Em especial agradeço ao Professor Doutor Sidney Carlos Ferrari por ouvir e apontar caminhos e sempre servir como agente motivador para esta jornada.

Agradecimentos especiais aos professores e colegas da Universidade Estadual de Londrina¹. Minha mente não cansa de se curvar em reverência diante de vocês. Vocês são o farol que indica o caminho para a jornada do saber.

Finalmente agradeço à Deus pela existência, à família pela paciência e aos amigos pela solidariedade.

¹ <<http://www.uel.br/cce/dc/>>

*“Não seja como o pescador que traz o peixe. Ensine a pescar.”
(de várias culturas e lugares)*

HYPÓLITO, João Maurício. **GAIA-SANM**: Framework para Gerenciamento de Projetos de Processos de Software. 69 p. Dissertação de Mestrado (Mestrado em Ciência da Computação) – Universidade Estadual de Londrina, Londrina, 2017.

RESUMO

Ter maturidade em projeto de software é apresentar o domínio sobre o desenvolvimento de artefatos e execução de processos que otimizem o gerenciamento de projeto de software. Maturidade em gerenciamento de projetos de software implica em obter um produto de melhor qualidade. Este trabalho tem como objetivo geral apresentar um ambiente onde seja possível avaliar o Nível de Maturidade de empresas desenvolvedoras de software. Nesse ambiente a software house se submete a uma avaliação e obtém como resposta um conjunto de sugestões de boas práticas em projetos. Essas melhorias permitem que seus processos de desenvolvimento de software evoluam, aumentando seu Nível de Maturidade. Para determinar o Nível de Maturidade se propõe uma escala de pontuação e um conjunto de pontos de referência para se estabelecer a estimativa. A escala de nível de maturidade é dividida em cinco patamares que devem ser alcançados pelos desenvolvedores de software através do desempenho de seus gerentes para as avaliações estruturadas. Este ambiente será estruturado em camadas segundo o padrão MVC e a aplicação final se apresenta em plataforma WEB. Justificando a eficácia do ambiente desenvolve-se uma simulação de avaliação de uma software house. Nessa simulação é apresentada a escala de pontuação, os pontos referenciais e as regras de determinação do Nível de maturidade.

Palavras-chave: Níveis de Maturidade. Gerenciamento de Projeto. Projetos de Software. Processos de Software. Frameworks para Gerenciamento.

HYPÓLITO, João Maurício. **GAIA-SANM**: Framework for Project Management of Software Processes. 69 p. Dissertation (Master's degree in Science in Computer Science) – Universidade Estadual de Londrina, Londrina, 2017.

ABSTRACT

Having maturity in software design is to present the domain on the development of artifacts and execution of processes that optimize the management of software design. Maturity in software project management implies getting a better quality product. This work has as general objective to present an environment where it is possible to evaluate the Maturity Level of software development companies. In this environment the software house undergoes an evaluation and obtains as answer a set of suggestions of good practices in projects. These improvements allow your software development processes to evolve, increasing your Maturity Level. To determine the Maturity Level, a scoring scale and a set of reference points are proposed to establish the estimate. The maturity level scale is divided into five tiers that must be achieved by software developers through the performance of their managers for structured evaluations. This environment will be structured in layers according to the MVC standard and the final application is presented in WEB platform. Justifying the effectiveness of the environment, a simulation of software house evaluation is developed. In this simulation is presented the scoring scale, the reference points and the determination rules of the Maturity Level.

Keywords: Maturity Levels. Project Management. Software Projects. Software Processes. Frameworks for Management.

LISTA DE ILUSTRAÇÕES

Figura 1 – Níveis de Maturidade.	21
Figura 2 – NM X Áreas de Gerenciamento - Artefatos.	27
Figura 3 – Família Gaia de Soluções de TI.	36
Figura 4 – Modelo entidade-relacionamento do Framework SANM.	39
Figura 5 – Diagrama de Caso de Uso: Administradores.	41
Figura 6 – Escala de Pontuação.	46
Figura 7 – Escala de Pesos.	47
Figura 8 – Diagrama de Caso de Uso: Desenvolvedora.	60
Figura 9 – Diagrama de Caso de Uso: Gerentes de Projeto de Software.	61
Figura 10 – Diagrama de Caso de Uso: Especialistas.	61
Figura 11 – Diagrama de Caso de Uso: Auditores.	62

LISTA DE TABELAS

Tabela 1 – Artefatos para o NM1 indicados por PMI	22
Tabela 2 – Artefatos para o NM2 indicados por PMI	22
Tabela 3 – Artefatos para o NM3 indicados por PMI	23
Tabela 4 – Artefatos para o NM4 indicados por PMI e COBIT	24
Tabela 5 – Artefatos para o NM5 indicados por PMI e COBIT	25
Tabela 6 – Áreas de Gerenciamento/Artefatos e NM associados	26
Tabela 7 – Item de Maturidade e Melhorias associadas para NM1	28
Tabela 8 – Item de Maturidade e Melhorias associadas para NM2	30
Tabela 9 – Item de Maturidade e Melhorias associadas para NM3	31
Tabela 10 – Item de Maturidade e Melhorias associadas para NM4	32
Tabela 11 – Item de Maturidade e Melhorias associadas para NM5	33
Tabela 12 – Tabelas da Base de Dados do SANM	40
Tabela 13 – Dados Simulados de uma resposta de avaliação para o NM2	44
Tabela 14 – Distribuição de Frequência de Pesos x Quantidade de Questões	47
Tabela 15 – Quadro Geral dos Níveis de Maturidade	49
Tabela 16 – Quadro Geral dos Níveis de Maturidade - Revisado	49
Tabela 17 – Dicionário de Dados do Framework SANM	54
Tabela 18 – Dados Simulados de Resposta de Avaliação para Todos os Níveis de Maturidade	63

LISTA DE ABREVIATURAS E SIGLAS

BD	Banco de Dados
CGI	Common Gateway Interface
COBIT	Control OBJECTives for Information and related Technology
CPM	Critical Path Method
CSS	Cascading Style Sheets
DFD	Diagrama de Fluxo de Dados
DHF	Diagrama Hierárquico Funcional
DicEAP	Dicionário de Termos da EAP
EAP	Estrutura Analítica do Projeto
FK	Foreign Key - Chave Estrangeira
HTML	HyperText Markup Language
ITIL	Information Technology Infrastructure Library
MER	Modelo Entidade-Relacionamento
MPS-BR	Melhoria em Processo de Software - Brasil
MVC	Model-view-controller
NM	Nível de Maturidade
PA	Programa(s) Aplicativo(s)
PERT	Program Evaluation and Review Technique
PGA	Plano de Gerenciamento de Aquisição
PGCO	Plano de Gerenciamento de Comunicação
PGC	Plano de Gerenciamento do Custo
PGE	Plano de Gerenciamento do Escopo
PGM	Plano de Gerenciamento de Melhorias
PGPI	Plano de Gerenciamento de Partes Interessadas

PGP	Plano de Gerenciamento do Projeto
PGQ	Plano de Gerenciamento de Qualidade
PGRH	Plano de Gerenciamento de Recursos Humanos
PGR	Plano de Gerenciamento do Risco
PGT	Plano de Gerenciamento do Tempo
PGV	Plano de Gerenciamento do Versionamento
PHP	Personal Home Page
PK	Primary Key - Chave Primária
PMBOK	Project Management Book of Knowledge
PMI	Project Management Institute
SANM	Sistema de Avaliação do Nível de Maturidade
SGBD	Sistema Gerenciador de Banco de Dados
TA	Termo de Abertura
TE	Termo de Encerramento
WWW	World Wide Web

SUMÁRIO

1	INTRODUÇÃO	14
2	NÍVEL DE MATURIDADE	18
2.1	Conceitos iniciais	18
2.2	Nível de Maturidade	19
2.3	Melhorias	27
3	FRAMEWORK: SANM - CARACTERIZAÇÃO	35
3.1	Introdução	35
3.2	O Framework SANM e o projeto GAIA	35
3.3	Especificação do SANM	37
3.3.1	A camada do banco de dados do SANM	37
3.3.2	A camada de controle do SANM	41
3.3.3	A camada de Interface do SANM	42
4	O CÁLCULO DO NÍVEL DE MATURIDADE	43
4.1	Condições iniciais e Dados	43
4.2	Pontuação	45
4.2.1	Parâmetros	45
5	CONSIDERAÇÕES FINAIS	50
	REFERÊNCIAS	52
	APÊNDICES	53
	APÊNDICE A – DICIONÁRIO DE DADOS DO SANM	54
	APÊNDICE B – CASOS DE USO DO SANM	60
	APÊNDICE C – SIMULAÇÃO DE UMA AVALIAÇÃO	63
	Trabalhos Publicados pelo Autor	69

1 INTRODUÇÃO

No período de 1960 a 2000 houve a evolução do hardware, o surgimento de ferramentas computacionais, novas metodologias para desenvolvimento de software, bem como a comunicação entre computadores através do uso da internet.

Desde 1960, as características dos computadores têm melhorado continuamente a uma taxa anual média de cerca de 20%. A aplicação dos transistores chegando aos circuitos integrados, tornou os computadores mais compactos, com maior poder computacional e a um custo cada vez mais baixo [1]. Em um primeiro momento os computadores eram direcionados para empresas, que precisavam de ferramentas para dar suporte na tomada de decisões. Segundo Magalhães [2], especialistas acreditam que as taxas de melhoria do hardware ainda se mantêm nos patamares das décadas de 1980 e 1990.

Vargas [3] indica que a partir da década de 1970 vê-se o crescimento no uso de ferramentas computacionais para suporte ao trabalho dos gerentes de projetos. Os programas da categoria CASE¹ inicialmente surgem para auxiliar na modelagem de sistemas e evoluem para duas direções distintas: geração de código e desenvolvimento de artefatos de gerenciamento de projetos [3]. Neste aspecto o gerente de projeto passa a contar com o apoio de software para auxiliá-lo na execução de suas atividades. Entretanto, para uma *software house* ainda faltam ferramentas que permitam melhorar a administração de projetos.

As metodologias de desenvolvimento de software inicialmente atribuíam qualidade ao produto de um projeto quando este atendia aos requisitos determinados ao produto [1, 4]. Depois passaram a acompanhar os processos produtivos do projeto com a intenção de atribuir qualidade aos produtos desenvolvidos. Este conceito evoluiu considerando em separado a qualidade do produto em detrimento da consideração do sucesso do projeto. Estas metodologias tratam do estudo e desenvolvimento do conhecimento em gerenciamento de projetos. Neste ambiente, a indústria de desenvolvimento de software encontrou um panorama favorável para acelerar continuamente passando a oferecer o desenvolvimento de sistemas de informação [2]. Sistemas computacionais permitiam um aumento significativo na qualidade da tomada de decisão no âmbito empresarial.

O surgimento da Internet, ao longo das décadas de 70 e 80, trouxe em seu repertório de serviços, um grupo de tecnologias que tornavam a computação cada vez mais próxima de um usuário que não necessariamente dominava os processos de desenvolvimento de software [4].

As Tecnologias da Informação e Comunicação (TICs) passam a oferecer três grupos

¹ CASE - *Computer-Aided Software Engineering*

de ferramentas que suportam o desenvolvimento de Sistemas de Informação: linguagens de programação, bancos de dados e formatadores de interfaces. Estes grupos de ferramentas podem ser usados em combinações diferentes para a construir sistemas. Estes métodos de trabalho passaram a ser alvo de estudo, buscando desenvolver software com melhor qualidade.

O estudo dos métodos de desenvolvimento de software permitiu o estabelecimento de padrões de qualidade no desenvolvimento de sistemas de informação e de software de modo geral. Surgia, então, um promissor mercado para o desenvolvimento de software. As empresas desenvolvedoras de softwares (*software houses*) passaram a ter como meta desenvolver programas com qualidade e a buscar a excelência no desenvolvimento de software.

Neste contexto, os administradores de *software house* rapidamente perceberam que o ritmo de desenvolvimento da qualidade de software somente poderia manter-se acelerado se fossem adotadas metodologias de gerenciamento de projetos. No âmbito das TICs surgem como metodologias de gerenciamento de projetos o ITIL², o COBIT³ e o MPS-BR⁴ como as principais propostas de organização de projetos de software. De modo mais amplo o PMI⁵ apresenta o PMBOK⁶, um compêndio de conhecimento em gerenciamento de projetos de amplo espectro, podendo ser aplicado ao contexto de projetos de software.

Para as *software houses*, alguns problemas surgiram e as soluções não são óbvias: é possível verificar se uma software house executa bons projetos software? Como identificar se a desenvolvedora de software tem processos de desenvolvimento que atendam a um determinado grupo de parâmetros que reconhecidamente balizam a qualidade da criação do software? Ou ainda: em uma instituição de ensino, como aferir o atendimento a um Padrão de Qualidade no desenvolvimento de software? Finalmente, como garantir a qualidade de um produto cuja construção é complexa e atendendo às especificações de um mercado cada vez mais exigente?

Neste contexto, pode-se propor o uso da TIC para desenvolver um sistema que permita avaliar uma *software house*, segundo pontos específicos de metodologias consagradas e de resultado comprovado, apontando melhorias que possam ser implantadas na *software house* avaliada.

O desenvolvimento de software é tratado como um projeto. O gerente do projeto é fundamental para o sucesso do projeto. Segundo PMI [5], "sucesso de um projeto é aferido quando sua conclusão se dá atingindo valores dentro de intervalos referenciais para parâmetros pré-determinados, tais como custo e prazo, além de desenvolver um produto

² ITIL - *Information Technology Infrastructure Library*

³ COBIT - *Control Objectives for Information and related Technology*

⁴ MPS-BR - *Melhoria em Processo de Software - Brasil*

⁵ PMI - *Project Management Institute*

⁶ PMBOK - *Project Management Book of Knowledge*

que atenda aos requisitos pré-estabelecidos pela parte interessada". As habilidades do Gerente de Projeto podem ser avaliadas pelo domínio apresentado no uso e criação de um conjunto de ferramentas que permitem concluir o projeto da maneira mais eficiente possível.

O PMI desenvolve o estudo sobre metodologias de gerenciamento de projeto e trata do conceito de maturidade, analisando a estrutura de execução do projeto. O gerenciamento de projetos é suportado por processos e artefatos sendo agrupados em dez áreas de gerenciamento, que são: gerenciamento de integração, escopo, tempo, risco, custo, qualidade, aquisição, recursos humanos, comunicação e partes interessadas [5]. Entretanto, o PMI não propõe uma escala de valores para a maturidade. Assim, este trabalho tem como objetivo geral definir os Níveis de Maturidade (NM) divididos em 5 níveis, quais sejam: conscientização, organização, acompanhamento, especialização e plenitude; apresentar uma estrutura que permita avaliar o NM de uma *software house* em desenvolvimento de software, indicando possíveis melhorias (boas práticas) que ajudem a *software house* a melhorar o gerenciamento de projetos de software.

Este objetivo geral pode ser atingido determinando-se os seguintes objetivos específicos:

- Apresentar o conceito de nível de maturidade em desenvolvimento de projetos de software;
- Apresentar uma escala de valores mensuráveis para o conceito de maturidade em projetos de software permitindo avaliar uma *software house*;
- Propor a estrutura de um *framework* de avaliação do (NM);
- Indicar um conjunto de melhorias que uma *software house* pode aplicar para evoluir seu nível de maturidade no desenvolvimento de software.

Este *framework* será parte dos produtos dentro do programa GAIA ⁷ desenvolvido no Departamento de Computação do Centro de Ciência Exatas da Universidade Estadual de Londrina.

O uso de TICs viabiliza a implementação deste *framework*, uma vez que é possível desenvolver software de qualidade usando pontos específicos de metodologias distintas. Desta forma o projeto GAIA pode construir o conhecimento que ajuda as *software houses* a gerenciar melhor seus projetos de software.

Este trabalho se justifica pelo fato de, atualmente, ser possível usar pontos específicos apresentados em metodologias distintas e conseguir o desenvolvimento de software com qualidade. É possível, desta forma, montar a escala do NM com áreas de gerenciamento de diversas metodologias de desenvolvimento de software.

Este trabalho tem a seguinte estrutura: capítulo 1: Introdução – apresenta-se um

⁷ laboratório GAIA - Soluções em TIC do Departamento de Computação da UEL

contexto e seu problema, uma proposta de solução, uma justificativa, objetivos gerais e específicos e a estrutura do trabalho; capítulo 2: Nível de Maturidade - Conceituação e caracterização e sua conexão com as avaliações às quais se submeterão as *software houses*; capítulo 3: FrameWork: SANM - Caracterização - especificação da estrutura do ambiente do framework que determina o NM e a maneira que é usada para determinar o NM; capítulo 4: Cálculo do Nível de Maturidade - apresentação do processo de determinação do NM de uma *software house*; capítulo 5: Análise de resultados - simulando avaliações para uma *software house*; capítulo 6 – Considerações finais - uma visão retrospectiva do desenvolvimento do Sistema de Avaliação do Nível de Maturidade (SANM); Referências - apresentam-se fontes de leitura e pesquisa; e Apêndices - tabelas complementares na estruturação do SANM e documentação do framework apresentado.

2 NÍVEL DE MATURIDADE

2.1 Conceitos iniciais

As metodologias desenvolvidas durante a década de 1980 atribuíam qualidade ao produto de um projeto quando este atendia aos requisitos determinados ao produto [1, 4]. Na década seguinte surge a ideia de se acompanhar os processos produtivos do projeto com a intenção de atribuir qualidade aos produtos desenvolvidos. Este conceito evolui considerando em separado a qualidade do produto em detrimento da consideração do sucesso do projeto.

Concluir um projeto com sucesso é determinar valores dentro de intervalos previamente estabelecidos para um grupo de parâmetros de controle [5]. Para isso é preciso não somente saber quais tarefas devem ser feitas para obter o produto, mas também quando e como executá-las. Entretanto existem situações que podem comprometer a execução das tarefas de um projeto, aumentando sua duração, seu custo ou até mesmo tornando sua execução impossível. Estas situações constituem um cenário de risco de execução e podem comprometer o sucesso de um projeto[6].

Segundo Dinsmore [1], o bom gerenciamento de projetos necessita que se saiba, com antecedência quais tarefas envolvem em sua execução uma maior condição de risco e qual a estratégia de correção na execução das tarefas, caso uma situação de risco ocorra. As habilidades de criar e manter a organização das ações necessárias para o bom desenvolvimento de um projeto podem ser englobadas em um termo: Maturidade.

O conceito de NM é abordado por Pressman [4] quando trata do assunto linha de base. A linha de base é um conjunto de dados que podem ser analisados e aumentam a maturidade no desenvolvimento e administração de projetos de software.

O PMI [5] trata do assunto maturidade, mas não propõe uma escala de valores. De modo complementar, COBIT [7], afirma o gerente de projetos de software deve, ainda, manter sua atenção no gerenciamento de mudanças (que pode criar a necessidade de gerenciar versões do produto desenvolvido). Nota-se também em COBIT [7] que o foco sobre maturidade está na busca constante por aperfeiçoamentos (através da manutenção de estruturas de auditorias) o que viabiliza a tomada de decisão mais complexa nos projetos, refletindo uma maior maturidade.

A estrutura de execução de um projeto é dividida em cinco fases: iniciação, planejamento, execução, monitoramento & controle e finalização (ou encerramento) [1, 3]. A partir deste ponto de vista COBIT [7], Magalhães [2] e PMI [5] tratam os projetos com subdivisões particulares. Entretanto, o conceito de NM não é abordado de maneira

explícita, nem é associado às fases principais de projetos em nenhuma das metodologias.

Os projetos são estudados e administrados a partir da determinação dos requisitos do produto final que será desenvolvido. No estudo de gerenciamento de projetos de softwares estes requisitos podem ser parâmetros referenciais que possibilitam avaliar a qualidade do produto que foi desenvolvido [8].

Entretanto, diferentes *software houses* podem construir produtos que atendam aos requisitos do produto; mas qual delas tem mais ou menos 'maturidade' em gerenciamento de projetos? A determinação da maturidade não pode, então, se basear só nas características do produto. O gerenciamento de projetos necessita do suporte de artefatos e da prática de processos que favoreçam a atividade de gerenciamento [5, 7].

2.2 Nível de Maturidade

O *Nível de Maturidade* (NM) é um índice que expressa o grau de conhecimento na determinação e desenvolvimento de artefatos de suporte ao gerenciamento do projeto bem como na implantação e execução de processos de gerenciamento do projeto. O NM permite, então, avaliar uma *software house* com uma métrica.

O NM forma uma escala progressiva e gradual, partindo de um estado onde se apresenta a conscientização de que o gerenciamento de projeto é importante para gerar registro e aprendizado dos projetos realizados, e terminando no estado onde o ato de gerenciar um projeto desenvolve todos os recursos que garantem que o gerenciamento aconteça com pleno sucesso. Os NMs são alcançados de modo sequencial e progressivo, pois o domínio sobre o uso de artefatos e processos de software obedece uma relação de dependência entre os níveis de conhecimento.

Neste trabalho os NM são apresentados em cinco estágios, denominados: Conscientização, Organização, Acompanhamento, Especialização e Plenitude, caracterizando-os segundo a existência de itens gerenciais. Cada um dos NM é identificado, respectivamente, por uma sigla, quais sejam: NM1, NM2, NM3, NM4, e NM5.

Os 5 NM são apresentados a seguir, enunciando-se sua sigla, sua identificação e a sua constituição em termos de artefatos, processos (requisitos de sua caracterização) e seus objetivos fundamentais.

- NM1: Conscientização.

Conceitos fundamentais de gerenciamento de projetos são conhecidos, porém as situações que levam ao desenvolvimento de um produto de baixa qualidade não são bem conhecidas. Neste NM existe a preocupação com 5 áreas de gerenciamento, quais sejam: integração, escopo, tempo, risco e custo. Entretanto, melhorias precisam ser implementadas para completar estas áreas de gerenciamento bem como o

desenvolvimento de outras áreas de gerenciamento. Neste nível de maturidade os conceitos fundamentais de gerenciamento existem mas precisam ser aperfeiçoados.

- NM2: Organização

Todos os conceitos de gerenciamento de projetos e uso de artefatos referentes ao NM1 são completos. Agora o gerenciamento é completado com artefatos e processos da área de gerenciamento de qualidade. A garantia de qualidade de processos passa a ser considerada no gerenciamento, o que reflete a maturidade na organização do projeto. A empresa sabe organizar as tarefas do projeto. As atividades deste nível de maturidade permitem intuir que a empresa, além de conscientização, tem uma noção do planejamento de ações para realizar as atividades do projeto, sabendo estudar a melhor forma de acompanhar a execução visando aumentar a qualidade do produto. A identificação do que será feito, como será feito e quando será feito são as principais características deste NM.

- NM3: Acompanhamento

Todos os conceitos de gerenciamento de projetos e uso de artefatos referentes aos níveis anteriores são completos. Agora o gerenciamento é completado com artefatos e processos da área de gerenciamento de comunicação, recursos humanos, aquisição e partes interessadas. O monitoramento e controle das atividades, a comunicação é mais efetiva, os recursos humanos e componentes externos são aplicados com maior eficiência e o acompanhamento das partes interessadas é efetivamente executado refletindo o uso apropriado de artefatos que permitem concluir um grau mais elevado de maturidade. A empresa consegue acompanhar a execução das tarefas de um projeto, embora ainda falte aprofundar o tratamento das situações que possam afetar o sucesso do projeto.

- NM4: Especialização

Todos os conceitos de gerenciamento de projetos e uso de artefatos referentes aos níveis anteriores são completos. Agora o gerenciamento é completado com artefatos e processos da área de gerenciamento de versionamento e melhorias contínuas. Observa-se no NM4 que a empresa, além de conscientização, organização e acompanhamento, exerce atividades que a tornam especialista na execução do projeto, podendo adotar medidas de tratamento das situações de risco. A empresa precisa, ainda, dominar as técnicas de desenvolvimento de soluções complementares ao produto do projeto (versões) e a desenvolver artefatos e processos novos que viabilizem o acompanhamento (auditorias) dos processos e artefatos desenvolvidos durante a execução do projeto.

- NM5: Plenitude

Todos os conceitos de gerenciamento de projetos e uso de artefatos referentes aos níveis anteriores são completos. No NM5 a empresa apresenta pleno domínio na execução do projeto. Neste nível de maturidade a empresa consegue analisar e desen-

volver constantemente novas melhorias que levam à implantação de novos processos e/ou desenvolvimento de novos artefatos de gerenciamento de projetos. A empresa tem a plenitude em gerenciamento de projetos.

Os Níveis de Maturidade podem ser apresentados de modo resumido na Figura 1, referenciando o aumento da capacidade de gerenciar projetos aliada ao aumento dos níveis de maturidade. Esta estratificação apresenta como uma empresa desenvolvedora de projetos 'evolui' em seu Nível de Maturidade; isso se dá à medida que implementa processo e desenvolve artefatos que dão suporte ao gerenciamento do projeto de software. A evolução se dá de maneira progressiva partindo do NM1 para o NM5, à medida que a empresa passa a desenvolver os itens de gerenciamento de projetos.

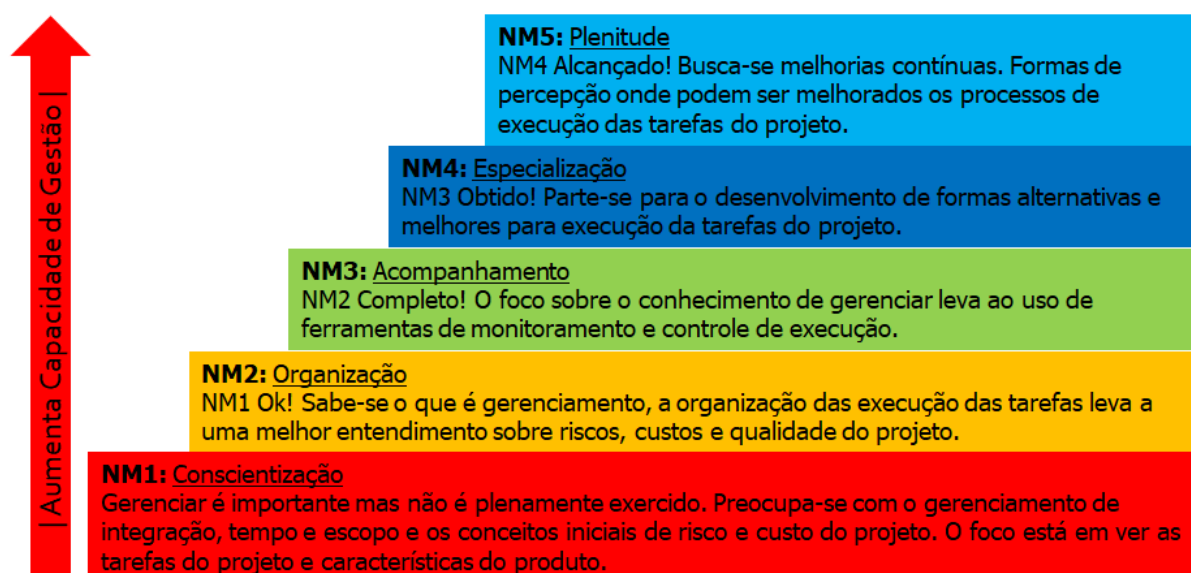


Figura 1 – Níveis de Maturidade.

Fonte: o autor

Neste trabalho os NM são associados aos artefatos que se tem para suporte no gerenciamento de projetos. Cada NM tem um conjunto de artefatos que devem ser obtidos e usados com maior rigor para melhor gerenciar um projeto, refletindo assim no melhor índice do nível de maturidade. Para cada área de gerenciamento, portanto, considera-se os artefatos de suporte e como eles são obtidos.

Estes artefatos, para o NM1, são apresentados na tabela 1. O NM1 tem seu foco principal na conscientização do gerenciamento de projeto como componente chave para o sucesso do projeto, destacando, portanto, artefatos das áreas de integração, escopo e tempo. Mas existe também neste NM o início da preocupação com riscos e custo do projeto. Os artefatos da tabela 1 são associados às áreas de gerenciamento apontadas em PMI [5] sendo apresentados com um nome e sigla e uma descrição.

Tabela 1 – Artefatos para o NM1 indicados por PMI

Área de Gerenciamento	Artefato	Descrição
Integração	Plano de Gerenciamento do Projeto (PGP).	Contém as diretrizes das outras áreas de gerenciamento do projeto.
	Matriz de Responsabilidade (MRS).	Define os perfis de profissionais/Pessoas que tem responsabilidade por tarefas no projeto.
	Termo de Abertura (TA).	Apresenta o contexto e características gerais do projeto.
Escopo	Plano de Gerenciamento de Escopo (PGE).	Detalha o contexto para o qual se executa o projeto
	Modelos do produto (MOD).	Representações do produto que será desenvolvido
Tempo	Plano de Gerenciamento de Tempo (PGT).	Define as estratégias de tratamento das tarefas do projeto ao longo do prazo de execução.
	Lista de Tarefas (LTAR).	Lista das Tarefas do projeto, indicando duração, precedência e sucedência de cada tarefa.
Riscos	Plano de Gerenciamento de Riscos (PGR).	Apresenta as diretrizes para tratamento das situações que podem comprometer o sucesso do projeto
Custos	Plano de Gerenciamento de Custos (PGC).	Define as estratégias de para determinar e acompanhar os custos e orçamentos do projeto.

A seguir, aos artefatos do NM1, são acrescentados os artefatos para as seguintes áreas de gerenciamento: integração, escopo, tempo, risco e qualidade, indicadas em PMI [5]. Estes artefatos caracterizam o NM2 e são apresentados na tabela 2.

Tabela 2 – Artefatos para o NM2 indicados por PMI

Área de Gerenciamento	Artefato	Descrição
Integração	Termo de Encerramento (TE).	Define os documentos e processos para declarar o fim do projeto.

Continua na próxima página

Tabela 2 – continuação da página anterior

Área de Gerenciamento	Artefato	Descrição
Escopo	Estrutura Analítica do Projeto (EAP).	Diagrama apresentando as divisões principais do projeto
	Dicionário da EAP - (DicEAP).	Apresenta explanação dos termos usados na EAP
Tempo	Rede PERT (<i>Program Evaluation and Review Technique</i>) [9](PERT).	Diagrama representando as tarefas, estados e encadeamento de execução do projeto.
	Gráfico de GANTT (GANTT).	Histograma indicando duração, limites de execução, caminho crítico e folgas das tarefas do projeto.
Risco	Registro dos Riscos (RGR).	Documento detalhando as tarefas e seus riscos associados
Custo	Planilha de Custos (PLC).	Planilha detalhando os custos das tarefas.
Qualidade	Plano de Gerenciamento de Qualidade (PGQ).	Artefato com as definições para tratamento da qualidade do projeto e do produto.

Obtido o domínio sobre os artefatos que caracterizam o NM2, pode-se apresentar os artefatos, recomendados por PMI [5], para as seguintes áreas de gerenciamento: comunicação, recursos humanos, aquisição e partes interessadas. Estes artefatos, somados aos artefatos dos níveis anteriores, caracterizam o NM3 e são indicados na tabela 3.

Tabela 3 – Artefatos para o NM3 indicados por PMI

Área de Gerenciamento	Artefato	Descrição
Risco	Planilha de Riscos (PLR).	Planilha para acompanhamento da execução das tarefas e seus riscos.
Qualidade	Planilha de Qualidade (PLQ).	Planilha para acompanhamento da realização das tarefas e sua adequação com os padrões de qualidade.
Comunicação	Plano de Gerenciamento de Comunicação (PGCO).	Documento com as diretrizes para tratamento das comunicações entre todas as esferas do projeto.
Continua na próxima página		

Tabela 3 – continuação da página anterior

Área de Gerenciamento	Artefato	Descrição
	Exemplo de Formulários (EFR).	Documento com a disposição de campos de formulários para entrada de dados comunicados no projeto.
Recursos Humanos	Plano de Gerenciamento de Recursos Humanos (PGRH).	Documento com as regras gerais para tratamento do recurso humano no projeto
Aquisição	Plano de Gerenciamento de Aquisição (PGA).	Documento com as diretrizes para gerenciamento das aquisições de serviços e/ou produtos de terceiros para o projeto.
Partes Interessadas	Plano de Gerenciamento de Partes Interessadas (PGPI).	Documento que define o tratamento de partes interessadas em acompanhar a execução do projeto.
	Planilha de Partes Interessadas (PPI).	Planilha para caracterizar as partes interessadas no projeto.

A seguir, aos artefatos dos níveis anteriores, são acrescentados os artefatos das seguintes áreas de gerenciamento: comunicação, recursos humanos, aquisição, partes interessadas recomendadas por PMI [5]; versionamento e melhorias contínuas recomendadas por COBIT [7]. Estes artefatos caracterizam o NM4 e são apresentados na tabela 4.

Tabela 4 – Artefatos para o NM4 indicados por PMI e COBIT

Área de Gerenciamento	Artefato	Descrição
Comunicação	Exemplo de Relatórios (ERL).	Documento com a especificação de layouts de relatórios de dados comunicados no projeto.
Recursos Humanos	Planilha de Alocação de Pessoal (PLP).	Planilha indicando os perfis de profissionais/pessoal e sua situação no projeto
Aquisição	Planilha de Fornecedores (PLF).	Planilha com os fornecedores de produtos/serviços para o projeto.
Partes Interessadas	Planilha de Acompanhamento das Partes Interessadas (PAP).	Planilha para acompanhar as tarefas de relacionamento com as partes interessadas.
Continua na próxima página		

Tabela 4 – continuação da página anterior

Área de Gerenciamento	Artefato	Descrição
Versionamento	Plano de Gerenciamento de Versões (PGV).	Documento definindo a forma de tratamento das versões do produto do projeto.
Melhorias Contínuas	Plano de Gerenciamento de Melhorias Contínuas (PGMC).	Documento com as diretrizes para determinação e implementação de melhorias.

Finalizando a declaração, na tabela 5 indicam-se os artefatos necessários e suficientes para, tendo completado todos os níveis anteriores, alcançar o NM5, obtendo a Plenitude em gerenciamento de projetos de software. As áreas de gerenciamento que incluem artefatos para este NM são: aquisição recomendada por PMI [5], versionamento e melhorias contínuas recomendadas por COBIT [7].

Tabela 5 – Artefatos para o NM5 indicados por PMI e COBIT

Área de Gerenciamento	Artefato	Descrição
Aquisição	Planilha de produtos Adquiridos (PLA).	Lista com os produtos adquiridos de terceiros para o projeto.
Versionamento	Exemplo de Pedidos de Mudanças (EPM).	Documento definindo campos e formatos para o pedido de mudanças nos requisitos de produtos do projeto.
Melhorias Contínuas	Planilha de Melhorias (PLM).	Planilha com as melhorias percebidas e sua situação no projeto.

Os dados indicados de cada área de gerenciamento que foram apresentados nas tabelas 1 até 5 podem ser recompilados na tabela 6. Desse modo, os artefatos do NM2 somam 17, os 9 artefatos do NM1 e mais os 8 que completam o NM2. Todos estes artefatos foram retirados das referências indicadas no nome das respectivas áreas de gerenciamento ou (quando for o caso) de alguma referência específica indicada no nome do artefato.

Na tabela 6 percebe-se que o NM2 tem todos os artefatos de NM1 e ainda completa as áreas de gerenciamento com mais artefatos, refletindo um aumento da maturidade em gerenciamento de projetos do NM1 para o NM2 e assim por diante.

Tabela 6 – Áreas de Gerenciamento/Artefatos e NM associados

Área de Gerenciamento	Artefato	NM1	NM2	NM3	NM4	NM5
Integração	PGP	X	X	X	X	X
	MRS	X	X	X	X	X
	TA	X	X	X	X	X
	TE		X	X	X	X
Escopo	PGE	X	X	X	X	X
	EAP		X	X	X	X
	DicEAP		X	X	X	X
	MOD	X	X	X	X	X
Tempo	PGT	X	X	X	X	X
	LTAR	X	X	X	X	X
	PERT		X	X	X	X
	GANTT		X	X	X	X
Risco	PGR	X	X	X	X	X
	RGR		X	X	X	X
	PLR			X	X	X
Custos	PGC	X	X	X	X	X
	PLC		X	X	X	X
Qualidade	PGQ		X	X	X	X
	PLQ			X	X	X
Comunicação	PGCO			X	X	X
	EFR			X	X	X
	ERL				X	X
Recursos Humanos	PGRH			X	X	X
	PLP				X	X
Aquisição	PGA			X	X	X
	PLF				X	X
	PLA					X
Partes Interessadas	PGPI			X	X	X
	PPI			X	X	X
	PAP				X	X
Versionamento	PGV				X	X
	EPM					X
Melhorias Contínuas	PGMC				X	X
	PLM					X

Os níveis de maturidade são associados aos artefatos das áreas de gerenciamento.

No contexto de desenvolvimento de software uma melhoria pode ser exemplificada deste modo: nos levantamentos de requisitos de software uma melhoria pode ser a implantação de um processo de auditoria do questionário de levantamento de requisitos. Esta melhoria no processo de levantamento não agrega valor diretamente ao produto, mas aumenta o padrão de qualidade do processo de levantamento de requisitos e indiretamente melhora o gerenciamento do projeto.

A execução de projetos pode ser dividida em áreas de gerenciamento. Na mesma referência ainda se indica quais são os artefatos e/ou processos que auxiliam cada área, então é possível associar aos itens de cada área de gerenciamento um conjunto de melhorias que podem ser implementadas para se alcançar um bom desenvolvimento dos artefatos e/ou processos [5].

No NM1, a empresa tem algum conceito de gerenciamento no contexto da execução do projeto. Entretanto, ainda falta dominar a execução de processos e/ou desenvolver artefatos que aumentem a maturidade aperfeiçoando o gerenciamento. Conhecer, aprender e aplicar estas melhorias é a regra que se aplica na evolução dos níveis de maturidade. Esta característica ocorre até o NM5, onde as melhorias completam a plenitude em gerenciamento de projetos.

O conceito de melhoria está referenciado em PMI [5] para projetos em um contexto geral e em COBIT [7] para o contexto de projetos de software. Ainda em PMI [5] algumas áreas de gerenciamento desenvolvem os artefatos de modo progressivo, então, também é possível indicar as melhorias a serem implantadas de modo evolutivo.

É possível, então, caracterizar uma tabela resumindo, para NM, os seus correspondentes artefatos e as melhorias associadas para alcançar o desenvolvimento destes artefatos. A tabela 7 tem os artefatos e melhorias para o NM1.

Tabela 7 – Item de Maturidade e Melhorias associadas para NM1

Item de Maturidade	Melhorias indicadas para obter o item
Plano de Gerenciamento do Projeto	Identificar, caracterizar e documentar os requisitos iniciais do projeto. Escolher quais as áreas de gerenciamento e seus principais artefatos necessários para o gerenciamento do projeto.
Matriz de Responsabilidade	Determinar artefatos e processos referenciais no projeto. Definir: responsáveis pelo desenvolvimento, pela revisão e aprovação dos artefatos/processos para o gerenciamento do projeto.
Continua na próxima página	

Tabela 7 – continuação da página anterior

Item de Maturidade	Melhorias indicadas para obter o item
Termo de Abertura.	Definir: responsável pela Parte Interessada e o Gerente do projeto. Determinar: requisitos gerais do produto, prazo de execução, orçamento total, condições restritivas e limites de aceitabilidade do produto.
Plano de Gerenciamento do Escopo	Documentar o levantamento de requisitos; indicar quais os modelos que serão usados no projeto; identificar os perfis dos profissionais envolvidos na modelagem; estabelecer processos de verificação dos modelos; atualizar a matriz de responsabilidade indicando novos artefatos desenvolvidos; atualizar todos os itens pertinentes e os novos itens no PGE.
Modelos do projeto	Analisar os requisitos; desenvolver os modelos indicados no PGE; atualizar todos os itens pertinentes e os novos itens no PGE.
Plano de Gerenciamento do Tempo	Determinar e documentar as diretrizes para o levantamento das tarefas do projeto, bem como seus tempos de execução e encaqueamento.
Lista de Tarefas	Determinar as tarefas do projeto, estimar sua duração e seu encaqueamento.
Plano de Gerenciamento de Riscos	Determinar e documentar as diretrizes para o levantamento das tarefas do projeto, bem como seus tempos de execução e encaqueamento.
Plano de Gerenciamento de Custos	Determinar e documentar as diretrizes para o levantamento das tarefas do projeto, bem como seus tempos de execução e encaqueamento.

É importante indicar que uma empresa pode estar fora da escala que determina o NM. Isso pode acontecer quando uma empresa não apresenta o mínimo conhecimento de técnicas de gerenciamento de projetos de software. Nesta situação o framework de avaliação contempla as questões respondidas com valor ZERO no seu índice.

Um importante aspecto do gerenciamento de projetos deve ser citado: a busca pelo aumento da maturidade exige que a *software house* desenvolva processos de revisão dos artefatos para adequação ao novo nível pretendido, mesmo ao se implantar as melhorias observadas para o desenvolvimento dos novos artefatos de gerenciamento. Portanto, para todos os NM pretendidos, os artefatos dos níveis anteriores devem ser revisados para sua

adequação ao nível. Este esforço aumenta a qualidade do artefato desenvolvido e otimiza o domínio gerencial da *software house*.

O NM2 é alcançado com o uso de outros artefatos (indicados na tabela 2). Desse modo pode-se indicar, para estes novos artefatos, quais são as melhorias que devem ser introduzidas no gerenciamento de projetos completando o NM2.

A tabela 8 indica, para o NM2, quais são as melhorias associadas aos itens de maturidade complementares ao NM1, concluindo o NM2.

Tabela 8 – Item de Maturidade e Melhorias associadas para NM2

Item de Maturidade	Melhorias indicadas para obter o item
Termo de Encerramento	Identificar os requisitos para aceitação das partes do produto. Determinar como coletar os aceites das partes do produto. Acompanhar a conclusão dos itens que determinam os aceites.
Estrutura Analítica do Projeto	Definir os módulos que compõem a EAP e suas correspondentes tarefas. Atualizar todos os itens pertinentes aos novos artefatos feitos (matriz de responsabilidade e plano de gerenciamento de escopo).
Dicionário da EAP	Identificar e definir termos usados na EAP que são particulares do projeto e descrevê-los. Avaliar o processo de desenvolvimento. Atualizar todos os itens pertinentes e os novos artefatos feitos (matriz de responsabilidade e plano de gerenciamento de escopo).
Rede PERT	Listar os estados do projeto e com a lista de tarefas indicando as precedentes e sucedentes de cada tarefa fazer a rede PERT. Determinar o <i>Critical Path Method</i> (CPM - método do caminho crítico). Calcular os tempos mais cedo e mais tarde para obtenção dos estados do projeto. Determinar as folgas de tempo para execução das tarefas.
Cronograma de GANTT.	Determinar o caminho crítico das tarefas do projeto. Montar o cronograma (Gantt). Encadear as tarefas em ordem cronológica e agrupar por módulos / submódulos da EAP.
Continua na próxima página	

Tabela 8 – continuação da página anterior

Item de Maturidade	Melhorias indicadas para obter o item
Registro de Riscos	Elaborar a lista de riscos (a partir da lista de tarefas) fazendo a análise qualitativa (chance de ocorrer, impacto e severidade do risco) e quantitativa (estimativa do custo e prazo de ajuste) dos riscos. Planejar respostas aos riscos. Agrupar os riscos por módulo da EAP. Definir o que fazer com os riscos para cada módulo (ou sub) da EAP. Definição de ações para evitar ou eliminar os efeitos de riscos nos módulos do projeto.
Plano de gerenciamento de qualidade	Documentar a estratégia de desenvolvimento da qualidade no projeto. Determinar as ações para garantir a qualidade nos processos do projeto. Determinar as formas de acompanhamento para o controle da qualidade. Registrar todos os itens no Plano de Gerenciamento de Qualidade.

O NM3 é alcançado com o uso de outros artefatos (indicados na tabela 3). Desse modo pode-se indicar, para estes novos artefatos, quais são as melhorias que devem ser introduzidas no gerenciamento de projetos completando o NM3.

A tabela 9 indica, para o NM3, quais são as melhorias associadas aos itens de maturidade complementares aos NM1 e NM2, concluindo o NM3.

Tabela 9 – Item de Maturidade e Melhorias associadas para NM3

Item de Maturidade	Melhorias indicadas para obter o item
Planilha de Riscos	A partir do registro de riscos montar a planilha de acompanhamento de execução das tarefas que tem riscos. Determinar a frequência do acompanhamento das tarefas de risco. Atualizar a Planilha com os dados levantados.
Planilha de Qualidade	Rever a lista de tarefas e o GANTT estabelecendo os momentos de medida do desenvolvimento das tarefas e sua adequação com os requisitos.
Plano de Gerenciamento de Comunicação	Determinar as formas de comunicação adequadas ao projeto. Determinar os leiautes de formulários e relatórios. Determinar a frequência de coleta de dados e geração de relatórios.
Continua na próxima página	

Tabela 9 – continuação da página anterior

Item de Maturidade	Melhorias indicadas para obter o item
Exemplos de Formulários	Escolha da métrica e periodicidade de acompanhamento de cada tarefa da planilha de acompanhamento de execução das tarefas do projeto.
Plano de Gerenciamento de Recursos Humanos	Determinar: os perfis de profissionais; a disponibilidade de pessoal (ativo organizacional ou contratado). Analisar as necessidades de pessoal ao longo da execução do projeto.
Plano de Gerenciamento de Aquisição	Determinar: componentes externos necessários no projeto; fornecedores de serviços ou componentes; formas de aquisição dos componentes externos; formas de pagar pela aquisição.
Plano de Gerenciamento de Partes Interessadas	Identificar as partes interessadas e determinar suas categorias como ativas ou passivas no projeto.
Planilha de Partes Interessadas	Planilhar os levantamentos feitos no Plano de Gerenciamento de Partes Interessadas.

O NM4 é alcançado com o uso de outros artefatos (indicados na tabela 4). Desse modo pode-se indicar, para estes novos artefatos, quais são as melhorias que devem ser introduzidas no gerenciamento de projetos completando o NM4.

A tabela 10 indica, para o NM4, quais são as melhorias associadas aos itens de maturidade complementares aos NM1, NM2 e NM3, concluindo o NM4.

Tabela 10 – Item de Maturidade e Melhorias associadas para NM4

Item de Maturidade	Melhorias indicadas para obter o item
Exemplo de Relatório.	Determinar os dados necessários em cada relatório do projeto. Montar o leiaute detalhado do relatório.
Planilha de Alocação de Pessoal.	Determinar origem dos perfis profissionais necessários no projeto. Montar o leiaute detalhado da planilha.
Planilha de Fornecedores.	Identificar dados dos fornecedores de componentes externos. Montar o leiaute detalhado da planilha.
Planilha de Acompanhamento de Partes Interessadas.	Determinar os dados necessários para acompanhar partes interessadas. Montar o leiaute detalhado da planilha.
Continua na próxima página	

Tabela 10 – continuação da página anterior

Item de Maturidade	Melhorias indicadas para obter o item
Plano de Gerenciamento de Versionamento.	Determinar os processos para gerenciamento de mudanças do produto. Identificar procedimentos e perfis profissionais envolvidos em mudanças do produto. Documentar pedidos. Documentar soluções adotadas aos pedidos de mudança. Definir agrupamento de mudanças (definir versões).
Plano de Gerenciamento de Melhorias Contínuas.	Determinar método para identificar artefatos obsoletos em gerenciamento de projetos. Determinar método para desenvolvimento de novos artefatos/processos de gerenciamento. Documentar desenvolvimento indicando vantagens dos novos artefatos/processos.

O NM5 é alcançado com o uso de outros artefatos (indicados na tabela 5). Desse modo pode-se indicar, para estes novos artefatos, quais são as melhorias que devem ser introduzidas no gerenciamento de projetos completando o NM5.

A tabela 11 indica, para o NM5, quais são as melhorias associadas aos itens de maturidade complementares aos NM1 até NM4, concluindo o NM5.

Tabela 11 – Item de Maturidade e Melhorias associadas para NM5

Item de Maturidade	Melhorias indicadas para obter o item
Planilha de Aquisição.	Determinar os fornecedores dos componentes externos. Identificar métodos de compra. Desenvolver a planilha para controle compras/recebimento dos componentes. Executar a compra/recebimento e manter planilha atualizada.
Exemplo de Pedidos de mudanças.	Determinar dados dos pedidos de mudanças. Montar leiaute do pedido de mudanças. Definir processo de desenvolvimento do pedido de mudanças.
Planilha de Melhorias.	Determinar os dados para acompanhamento do uso de artefatos e processos. Determinar o leiaute da planilha de melhorias. Definir método de uso da planilha.

Foram apresentados até aqui os conceitos que fundamentam o framework para gerenciamento de projetos de processos de software. Foram apresentados e analisados:

Nível de Maturidade, projeto de software e áreas de gerenciamento de projetos de software, artefatos de suporte ao gerenciamento e melhorias em projeto de software, que visam aumentar a eficiência na gerência de projetos e conseqüentemente na qualidade do produto obtido, bem como os conceitos de transição entre níveis de maturidade.

Isso conclui a descrição dos conceitos necessários para a fundamentação do framework.

3 FRAMEWORK: SANM - CARACTERIZAÇÃO

Apresentados os conceitos básicos para a fundamentação do framework, neste capítulo formaliza-se a especificação da estrutura do framework indicando-se os recursos de TIC necessários para seu desenvolvimento.

3.1 Introdução

Um framework (computacional ou não) é um ambiente onde é possível se adotar uma metodologia de trabalho para gerenciamento de projetos [3, 5]. Este ambiente pode ser implementado em pasta, folhas de papel e arquivos físicos. Este ambiente também pode ser construído em meios computacionais. Neste capítulo apresenta-se uma estrutura para o desenvolvimento de um framework para gerenciamento de projetos de software em um meio computacional.

O framework do Sistema de Avaliação do Nível de Maturidade (SANM) é baseado em um Banco de Dados (BD), em programas aplicativos (PAs) e em um conjunto de interfaces. As telas são controladas pelos PAs e oferecem aos usuários o acesso aos dados do BD. Além destes serviços, os PAs implementam as regras de controle para determinação dos Níveis de Maturidade de uma *software house*. A partir da execução de uma avaliação o SANM determina o NM e apresenta as melhorias de gerenciamento recomendadas para o aprimoramento do ciclo de desenvolvimento de software.

3.2 O Framework SANM e o projeto GAIA

O Laboratório GAIA que tem como objetivo propor soluções inovadoras para apoio e realização de projetos de TIC. Estas soluções envolvem diversos modelos e técnicas nas áreas de Engenharia de Software, Gerenciamento de Serviços de TI e Governança de TI. Basicamente, a Família Gaia de Soluções de TI divide-se em Frameworks, Softwares, Processos e Objetos de Aprendizagem, como podemos observar na Figura 3.

Dentre os vários trabalhos realizados no âmbito da Família GAIA, tem-se:

- GAIA-Governança de TIC: framework para avaliar a governança de TIC nas organizações de maneira objetiva e oferecer diretrizes para o seu desenvolvimento.
- GAIA-Estimativa: um framework para gerência e avaliação das práticas de estimativas de software. Este framework foi desenvolvido com o objetivo de apoiar a gerência e avaliação do processo de estimativa de software, coordenando efetivamente os projetos de desenvolvimento.

- GAIA-Riscos: framework para o gerenciamento de riscos no processo de desenvolvimento de software. Este framework foi desenvolvido para facilitar a adoção de ações para aderir aos padrões de gerenciamento de riscos no processo de desenvolvimento de software e permitir a implantação gradativa e incremental de gerência, aumentando a qualidade do software gerado.
- GAIA-Lições Aprendidas: framework desenvolvido com o objetivo de aplicar a prática de lições aprendidas de forma gradativa e incremental dentro do processo de desenvolvimento de software, aumentando a qualidade do software gerado e consequentemente o conhecimento dentro da empresa.
- GAIA-Catálogo de Serviços: framework para facilitar a implantação de um catálogo de serviços em qualquer tipo de organização. Esta implantação é realizada de maneira gradativa e incremental, buscando atender as diretrizes do ITIL.
- GAIA-Regras de Negócio: possui a finalidade de aperfeiçoar o gerenciamento das regras de negócio visando elevar o nível de qualidade dos softwares e garantir o gerenciamento das necessidades reais em negócio, antes da definição dos requisitos de software. O framework define um procedimento de implantação baseado em um questionário de avaliação diagnóstica que identifica o nível de maturidade institucionalizado na organização

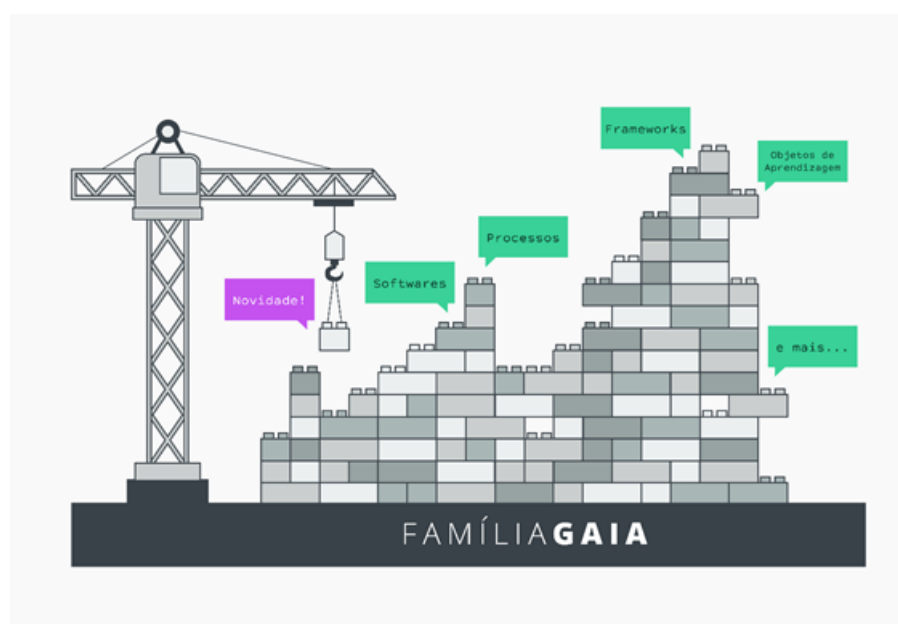


Figura 3 – Família Gaia de Soluções de TI.

Fonte: <http://www.gaia.uel.br>

Na página da Família GAIA (<http://www.gaia.uel.br>), podem ser encontrados outros frameworks, softwares, processos e objetos de aprendizagem que auxiliarão empresas na melhoria contínua de suas atividades.

3.3 Especificação do SANM

A arquitetura de aplicativos MVC¹ prevê que um sistema tenha três camadas: a camada de interface, a camada de controle e a camada do banco de dados. O SANM foi projetado para ser executado na plataforma *World Wide Web*². A Interface é controlada pelo Apache³ e implementada em HTML⁴/CSS⁵, a camada CGI⁶ em PHP⁷ e camada do SGBDR⁸ com o PostgreSQL⁹.

Esta arquitetura de programas foi implementada em dois ambientes distintos: Produção e Desenvolvimento. Em ambos todos os programas foram instalados e configurados para trabalhar na mesma forma. Estes ambientes diferem na configuração do equipamento e no software básico disponibilizado. No ambiente de produção foi usado uma versão do Linux Ubuntu, que opera o servidor de página, o interpretador PHP e o servidor de banco de dados PostgreSQL da mesma forma que o equipamento de desenvolvimento. No equipamento de desenvolvimento foi usado o sistema operacional Windows 10.

Um item de diferença entre os equipamentos foi a escolha dos editores de programas, instalados somente no ambiente de desenvolvimento. Neste foram adquiridos e instalados os editores Notetab Pro 6.0, o editor Sublime Text 3 e o editor Notepad++. Estes três editores facilitaram a edição, testes e depuração dos programas que constituíram a primeira implantação do SANM.

A aplicação é executada em duas camadas: Servidor e Cliente. Na camada cliente é executado o navegador, que se comunica por uma rede com o servidor onde solicita ao servidor de página a execução do PA (escrito em PHP). O Apache (servidor) então negocia com o interpretador PHP a execução dos comandos que constroem as interfaces em HTML/CSS ou acessam os bancos de dados controlados pelo PostgreSQL. Ao acessar o banco de dados o PA monitora a situação da solicitação (consulta ou atualização) e envia o tratamento desta resposta ao Apache no formato de comandos em HTML/CSS formatando a interface de resposta.

3.3.1 A camada do banco de dados do SANM

O banco de dados do SANM foi projetado com os princípios de modelagem de sistemas usando o Modelo Entidade-Relacionamento (MER), entretanto, como o SGBD do projeto é o PostgreSQL (Relacional), optou-se por apresentar os relacionamentos com

¹ MVC - Model-View-Controller.

² *World Wide Web*(WWW) - sistema hipertextual que opera através da internet.

³ Apache - Servidor de páginas Web

⁴ HTML - HyperText Markup Language

⁵ CSS - Cascading Style Sheets

⁶ CGI - Common Gateway Interface

⁷ PHP - Personal Home Page.

⁸ SGBDR - Sistema Gerenciador de Banco de Dados Relacional

⁹ PostgreSQL - SGBDR baseado em tabelas, desenvolvido em código aberto desde 1986.

cardinalidade N:M já desmembrados em entidades e relacionamentos com cardinalidade máxima 1:N. Esta forma de representação torna a especificação do banco de dados mais facilmente implementada em SGBDR. Desse modo o Banco de Dados implementado está representado na figura 4.

O diagrama da Figura 4 apresenta a indicação de cardinalidades máximas e mínimas, podendo-se abstrair algum comportamento de regras de negócio (pelo menos as que expressam o comportamento de chaves estrangeiras). Algumas regras de negócio do comportamento das integridades referenciais foram implementadas com o uso de regras de restrição (constraints, para o PostgreSQL).

O SANM aceita que as questões sejam referenciadas a um processo e/ou artefato das áreas de gerenciamento de projetos de software. As perguntas são elaboradas por especialistas nas áreas de gerenciamento de projeto de software e devem ser formadas com texto simples e objetivo, prevendo que a resposta seja indicada em uma escala de zero à cinco.

Para cada questão construída o especialista deve indicar à qual NM a questão está associada dentro da área de gerenciamento de sua especialidade. Ao elaborar a questão o especialista deve atribuir um peso de importância para a questão perante os conceitos tratados na área de gerenciamento. O SANM não implementa nenhum algoritmo que questione esta atribuição, no entanto usa estes valores para determinar o NM de uma empresa que submete uma avaliação.

O modelo indica que cada questão pode ser escrita por um especialista. Cada questão também está associada a um artefato que por sua vez se relaciona com áreas de gerenciamento. Ao elaborar uma questão o especialista atribui um peso à questão, este peso é um grau de importância do artefato associado à questão no contexto da área de gerenciamento considerada no NM.

Uma questão é relacionada a um só artefato do gerenciamento de projeto de software. Desta forma um gerente, ao responder uma questão está demonstrando seu conhecimento e uso de um artefato e reconhecendo sua importância no gerenciamento do projeto. O especialista pode desenvolver uma lista de melhorias para associá-las às questões. Estas melhorias apresentam um texto explicando o que é a melhoria e como implementá-la no gerenciamento de projeto, e a seguir permite que esta melhoria seja apontada para uma ou mais questões.

A conexão entre questão e melhoria se baseia no valor da resposta que será indicada pelo gerente. No registro da melhoria se apresenta um atributo denominado valência que tem o significado de indicar com qual valor de resposta a melhoria deve estar conectada. O valor da valência conecta a melhoria a uma resposta com valor menor ou igual ao seu valor. Por conta disso o valor da valência tem valor máximo 4, uma vez que respostas com

a base de dados do SANM. Os campos de cada tabela são apresentados no Apêndice A na tabela 17 onde descrevem-se seus tamanhos, tipos de dados, ordem (na tabela) e descrição semântica do seu conteúdo.

Tabela 12 – Tabelas da Base de Dados do SANM

Seq.	Ent-Tab	Descrição
1	administradores	Tabela com os administradores do SANM.
2	areadegestao	Área de gestão é uma divisão do grupo de processos de gerenciamento de projetos.
3	artefatos	Artefatos que permitem a obtenção do NM a partir de uma avaliação de uma <i>software house</i> .
4	avaliacaoquestao	Questões que compuseram uma avaliação de NM de uma <i>software house</i> .
5	avaliacoes	Tabela com a lista de avaliações de NM respondidas pelos gerentes das <i>software houses</i> .
6	idades	Tabelas com as cidades onde estão as desenvolvedoras, os especialistas ou os gerentes de projetos.
7	desenvolveagestao	Ligação entre desenvolvedoras e áreas de gestão.
8	desenvolvedoras	Tabela com as empresas que desenvolvem projetos de software.
9	especialistas	Especialistas de uma área de gestão que elaboraram questões para avaliação do NM.
10	gerentes	Gerentes de desenvolvedoras que respondem a avaliação do NM.
11	melhoriaquestao	Ligação entre as melhorias dos projetos de software e as respostas das questões das avaliações.
12	melhorias	Melhorias propostas pelos especialistas para desenvolver os artefatos que aprimorem o gerenciamento de projetos de software.
13	niveisdematuridade	Tabela com os níveis de maturidade percebidos no gerenciamento de projeto de software.
14	questoes	Tabela com as questões apresentadas no formulário de avaliação do NM em desenvolvimento de Software.

3.3.2 A camada de controle do SANM

Para a especificação da camada de controle, onde se apresentam os PA, usou-se o diagrama de Caso de Uso da UML-2.0¹⁰. O SANM possui quatro atores: Administradores, Desenvolvedoras, Gerentes de projetos (das desenvolvedoras) e os Especialistas.

Os PAs têm como foco básico permitir que uma *software house* responda a uma série de questões e receba o índice do seu nível de maturidade no desenvolvimento de software (NM1 até NM5). Quando terminar de responder uma avaliação e tiver o NM determinado, o gerente pode buscar as sugestões de melhorias para o gerenciamento de projeto de software que estão referenciados no SANM.

O diagrama de Caso de Uso dos Administradores é apresentado na Figura 5. O Administrador do SANM tem o papel de super usuário sobre os dados do sistema. Pode, através das funcionalidades disponíveis, atuar sobre todos os dados do SANM. Os diagramas foram desenvolvidos com o programa Visio¹¹ usando as bibliotecas da UML-2.0. Todos os diagramas foram feitos pelo autor do trabalho.

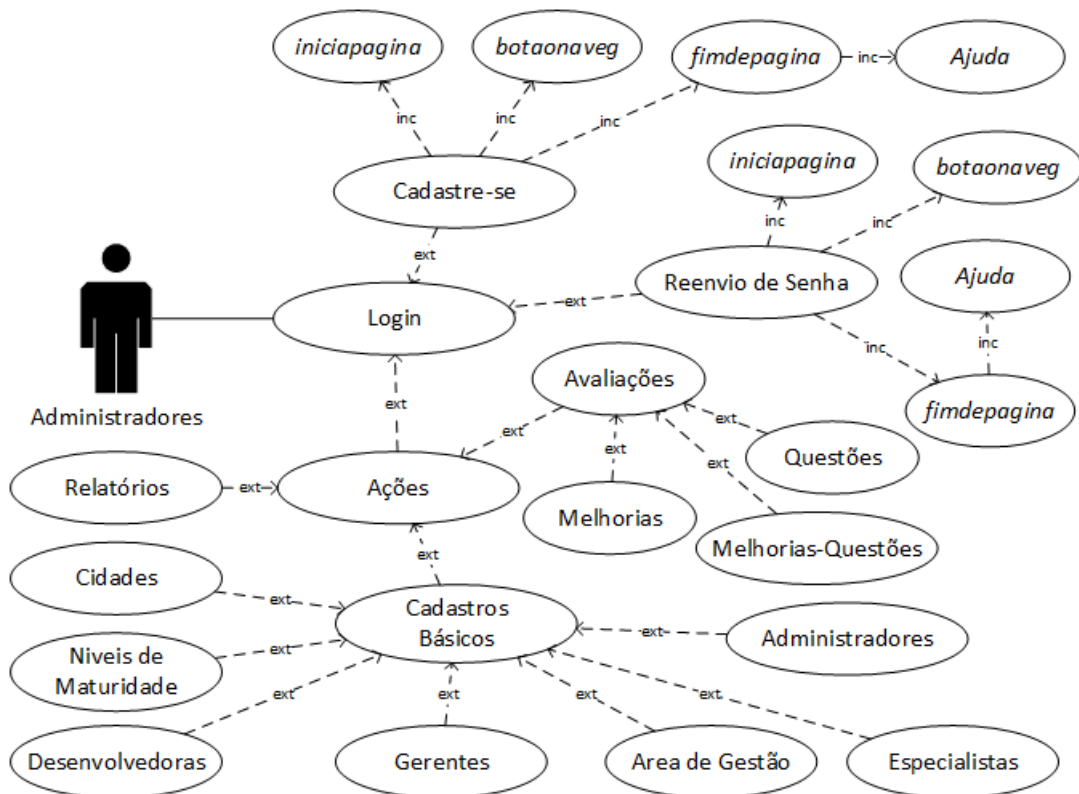


Figura 5 – Diagrama de Caso de Uso: Administradores.

Fonte: o autor

¹⁰ UML-2.0 - Linguagem de Modelagem Unificada (do inglês, UML - Unified Modeling Language) é uma linguagem-padrão para a elaboração da estrutura de projetos de software

¹¹ Visio - Aplicativo para criação de diagramas para o ambiente Windows.

Optou-se por apresentar em cada PA um cabeçalho elucidando referências de autoria, identificação do PA além de objetivos e descrições, bem como datas de criação e atualização do arquivo. Também fazem parte da documentação o algoritmo do PA e os comentários desenvolvidos ao longo do código-fonte implementado que orienta e explica as alternativas de tecnologia da linguagem que foram usadas na implementação do PA.

3.3.3 A camada de Interface do SANM

O SANM tem a camada de interface construída com HTML sempre dividida em três partes implementadas com o uso da tag <DIV>. Isso ocorre em todos os programas de manutenção dos dados de cadastros, com exceção das emissões de relatórios ou listagens de dados. As divisões superior e inferior são fixas em relação à borda da região de visualização. A parte central da tela fica com uma divisão de altura e largura variáveis. Isso é possível com aplicação de instruções de CSS na declaração das divisões (declaradas com o comando <DIV> da HTML).

As interfaces são codificadas em programas escritos em PHP de tal forma que quando executados constroem, em tempo de execução, as interfaces em HTML. Esta técnica de programação permite que o programa PHP receba variáveis e, desta forma, possa tornar a codificação HTML sujeita ao dinamismo do valor das variáveis. Isso é construção de programas com conteúdo dinâmico.

A linguagem PHP permite o desenvolvimento de programas estruturados na seguinte forma: programas principais, funções e subprogramas.

Os programas principais, em geral, estão associados à funcionalidades apresentadas nos diagramas de caso de uso. Funções são estruturas usadas para declarar segmentos de códigos cujo uso frequente (e alto) e é acionado a partir de um programa principal. Um programa principal e suas funções são escritas no mesmo arquivo de programa (arquivo texto). Um subprograma é um segmento de código que pode ser acionado a partir de mais de um programa principal, um subprograma, no entanto, pode conter segmentos de códigos implementados como funções.

O uso de funções acelera a execução do código, uma vez que todo o segmento de função fica em memória facilitando o controle de paginação de execução de programas por parte do interpretador do PHP. Por esta característica optou-se por usar a propriedade de *funções* para se escrever subprogramas e funções no SANM.

Esta característica da linguagem permite a estruturação das funcionalidades e funções do sistema em arquivos de programas. A linguagem também permite a construção de PA em modo recursivo, o que diminui a quantidade de arquivos acessados para a execução de uma funcionalidade do sistema.

4 O CÁLCULO DO NÍVEL DE MATURIDADE

Este capítulo tem como objetivo demonstrar o modo de cálculo para a determinação do NM de uma *software house*. Uma empresa se submete a uma avaliação que é composta de questões. Seus gerentes (se houver mais de um na software house) indicam suas respostas de cada questão e a partir daí se inicia a determinação do NM.

4.1 Condições iniciais e Dados

Os especialistas ao formularem as questões atribuem, para cada questão, um peso variando de 1 até 5. Este peso considera o grau de dificuldade da questão e/ou a importância do artefato de gerenciamento tratado no conteúdo da questão. As questões verificam o domínio ou conhecimento dos artefatos das áreas de gerenciamento ligadas ao NM.

Os gerentes de projetos respondem as questões indicando em uma escala de 0 até 5 o seu conhecimento ou domínio sobre o uso (ou desenvolvimento) de artefatos de gerenciamento. As questões de uma avaliação são associadas às áreas de gerenciamento propostos pelo PMI [5] e COBIT [7]. Quando uma resposta é assinalada com 0 (zero) significa que o conceito tratado na questão não é do domínio da empresa (através de seus gerentes). O conceito abordado na questão é relativo à um artefato quanto à sua definição ou uso no gerenciamento de projetos.

Para descrever os parâmetros e o cálculo do NM desenvolveu-se um grupo de dados simulando as respostas obtidas em uma avaliação. Estes dados são apresentados na tabela 13. Nesta tabela são indicados os níveis de maturidade na coluna NM, as questões correspondentes ao NM indicado na coluna Questão, os pesos indicados pelos especialistas para cada questão na coluna Peso, a resposta apontada por um gerente para cada questão na coluna Resposta e os pontos alcançados na questão na coluna Pontos. Os dados simulados apresentados na tabela 13 são referentes ao *NM2*.

Tabela 13 – Dados Simulados de uma resposta de avaliação para o NM2

NM	Questão	Peso	Resposta	Pontos
2	1	1	4	4
2	2	2	5	10
2	3	1	4	4
2	4	2	5	10
2	5	3	4	12
2	6	4	4	16
2	7	5	5	25
2	8	5	5	25
2	9	4	4	16
2	10	5	5	25
2	11	4	4	16
2	12	5	5	25
2	13	3	4	12
2	14	4	5	20
2	15	5	4	20
2	16	2	3	6
2	17	3	5	15
2	18	4	4	16
2	19	5	5	25
2	20	3	3	9
2	21	4	5	20
2	22	3	4	12
2	23	4	5	20
2	24	5	3	15
2	25	3	4	12

4.2 Pontuação

Tomando os dados da tabela 13 apresenta-se o conceito de pontuação da questão e os parâmetros P_{min} , P_{max} , P_{alc} e P_{lim} .

Os pontos alcançados em cada questão respondida são calculados com a seguinte fórmula:

$$Ponto = Peso * Resposta \quad (4.1)$$

Onde:

Ponto Pontuação da *Questão* com base no *Peso* e *Resposta*.

Peso Peso indicado pelo especialista para cada questão. O valor varia de 1 até 5.

Resposta Resposta indicada pelo gerente para cada questão. O valor varia de 0 até 5.

O cálculo dos pontos é aplicado em todos os dados simulados para o cálculo do NM e é apresentado no Apêndice C na tabela 18, do mesmo modo que definido na tabela 13.

4.2.1 Parâmetros

A partir dos dados da tabela 13 e da fórmula de pontuação, determinam-se os parâmetros:

- Pontuação alcançada:

É determinada pela aplicação direta da fórmula de pontuação 4.1. Como cada NM_i tem um grupo de k questões, a fórmula que determina a pontuação alcançada (P_{alc_i}) em uma avaliação para um determinado segmento NM_i pode ser escrita assim:

$$P_{alc_i} = \sum_{j=1}^k Peso_{i,j} * Resp_{i,j} \quad (4.2)$$

Onde:

i índice assumindo os valores de 1 até 5, indicando os níveis de maturidade.

j índice assumindo os valores de 1 até k , indicando a quantidade de questões.

k quantidade de questões para cada NM_i .

P_{alc_i} Pontuação alcançada para as k questões de um NM_i .

$Peso_{i,j}$ Peso de cada questão das k questões de NM_i , valor varia de 1 até 5.

$Resp_{i,j}$ Resposta de cada questão das k questões de NM_i , valor varia de 0 até 5.

Para os dados da tabela 13 a pontuação alcançada (P_{alc_2}) é: 390.

- Pontuação mínima:

Se todas as questões do segmento NM_2 forem respondidas com 0 (zero) a pontuação calculada aponta para seu valor mínimo. Assim:

$$P_{min_i} = \sum_{j=1}^k Peso_{i,j} * 0 = 0(\text{zero}) \quad (4.3)$$

- Pontuação máxima:

Se todas as respostas forem indicadas com resposta 5 (completo domínio sobre o conceito da questão), o valor calculado aponta para o valor máximo. Assim escreve a equação 4.4:

$$P_{max_i} = \sum_{j=1}^k P_{eso_{i,j}} * 5 = 5 * \sum_{j=1}^k P_{eso_{i,j}} \quad (4.4)$$

Para os dados da tabela 13, temos que $\sum_{j=1}^k P_{eso_{i,j}} = 89$, portanto, a pontuação máxima (P_{max_2}) é: 445 .

Com estes três parâmetros definidos se caracteriza, para cada segmento NM_i , a *Escala de Pontuação*: [P_{min_i} , P_{alc_i} e P_{max_i}], esquematicamente representada na figura 6.

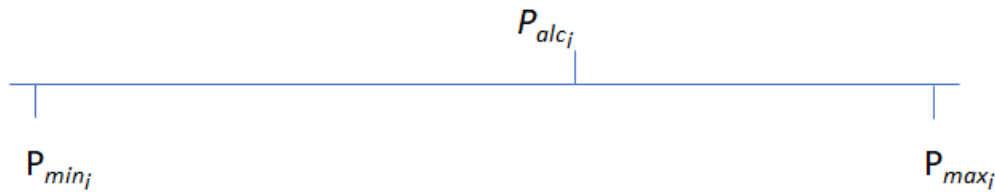


Figura 6 – Escala de Pontuação.

Fonte: o Autor

O valor de P_{alc_i} reflete o domínio no uso dos conceitos de gerenciamento abordados na k questões do NM_i , mas o quanto próximo de P_{max_i} deve estar P_{alc_i} para dizer que NM_i foi alcançado? Para responder satisfatoriamente esta pergunta P_{alc_i} deve ser comparado com um valor que caracteriza o grau de dificuldade das k questões do NM_i . É preciso, então, apresentar uma escala de valores que represente uma grandeza cuja magnitude seja diretamente proporcional aos valores dos pesos das questões. Sendo estes pesos os mesmos avaliados nas respostas apresentadas pelos gerentes que respondem à uma avaliação.

A Estatística indica uma grandeza que reflete o peso médio da distribuição da quantidade de questões por peso, das k questões de um NM_i . Esta grandeza é a média da distribuição de questões por Peso (MDF). Para determinar o valor desta grandeza determina-se, para cada grupo de k questões de um NM_i , a quantidade de questões para cada valor de peso.

Como exemplo tem-se os dados da tabela 13 e nela faz-se a contagem da quantidade de questões para cada peso. Desenvolve-se, assim, uma distribuição de frequência das questões por peso. Os dados deste cômputo são apresentados na tabela 14. Nesta tabela a coluna QQuest é a quantidade de questões correspondente a cada peso do NM da avaliação apresentada na tabela 13. Nota-se que a distribuição de questões por peso, apresentada nesta tabela, indica que existe uma quantidade de questões com peso 4 e 5, indicando

quais artefatos neste nível são importantes, uma vez que, em 14 das 25 questões, os pesos assinalados pelos especialistas são 4 e 5.

Tabela 14 – Distribuição de Frequência de Pesos x Quantidade de Questões

Peso	QQuest
1	2
2	3
3	6
4	7
5	7

Esta distribuição de frequência tem uma média da distribuição de frequência de peso calculada pela fórmula:

$$MDF_{NM_i} = \frac{\sum_{p=1}^5 \text{Peso}_{ip} * \text{QQuest}_{ip}}{\sum_{p=1}^5 \text{QQuest}_{ip}} \quad (4.5)$$

Onde:

i índice assumindo os valores de 1 até 5, indicando os níveis de maturidade.

p índice assumindo os valores de 1 até 5, indicando os pesos e as correspondentes quantidades de questões.

MDF_{NM_i} Média da distribuição de frequência de questões por peso para o NM_i .

Peso_{ip} pesos das questões de cada NM_i , com valores de p de 1 até 5, .

QQuest_{ip} quantidade de questões de cada peso no intervalo de questões de um NM_i .

Nos dados apresentados na tabela 13 o processamento da totalização de questões por NM_i para o NM exemplificado leva à construção da tabela 14. Pelo desenvolvimento, agora, do MDF_{NM_i} , chega-se ao valor: $MDF_{NM_2} = 3.56$.

Os pesos formam uma *Escala de Pesos* onde o MDF_{NM_i} é um ponto que expressa o grau de dificuldade média das questões. Pode-se representar esquematicamente esta escala de valores na figura 7.

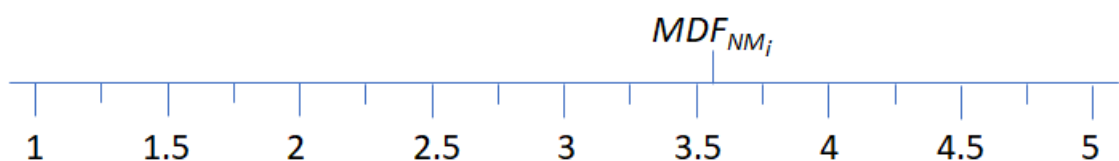


Figura 7 – Escala de Pesos.

Fonte: o autor

O peso da questão é, portanto, a grandeza referencial que liga a escala de peso e a escala de pontuação. Estas escalas apresentam o mesmo comportamento de proporcionalidade. Na escala de pontuação existe um ponto que corresponde ao MDF_{NM_i} , este é o P_{lim_i} . O P_{lim_i} é o ponto da escala de pontuação acima do qual se diz que o NM_i é alcançado. A regra de proporcionalidade entre MDF_{NM_i} e P_{lim_i} é expressa pela equação 4.6, na forma:

$$\frac{P_{max_i}}{5} = \frac{P_{lim_i}}{MDF_{NM_i}} \quad (4.6)$$

Desenvolvendo a equação 4.6, usando a equação 4.4, temos a expressão final que determina o P_{lim_i} , na forma da equação 4.7:

$$P_{lim_i} = MDF_{NM_i} * \sum_{j=1}^k P_{eso_{i,j}} \quad (4.7)$$

Onde:

- i índice indicando os níveis de maturidade, assume valores de 1 até 5.
- j índice indicando os pesos das questões, assume valores de 1 até k .
- P_{lim_i} Valor correspondente à MDF_{NM_i} na escala de pontuação da avaliação.
- MDF_{NM_i} Média da Distribuição de Pesos por Quantidade de Questões do segmento NM_i .
- $P_{eso_{ij}}$ pesos das questões de NM_i , com valores de p de 1 até 5.

Com os dados apresentados na tabela 13 o processamento da equação 4.7 leva à determinação do valor de $P_{lim_2} = 316.84$.

Com base nos valores da Escala de Pontuação (P_{alc_i} , P_{lim_i} e P_{max_i}) e no valor de P_{lim_i} , tem-se a regra I que determina o NM no segmento i na seguinte forma:

$$\begin{aligned} \text{SE } P_{alc_i} < P_{lim_i} & \text{ ENTÃO } NM_i & \text{ Insuficiente} \\ \text{SE } P_{alc_i} \geq P_{lim_i} & \text{ ENTÃO } NM_i & \text{ Suficiente} \\ \text{SE } P_{alc_i} = P_{max_i} & \text{ ENTÃO } NM_i & \text{ Completo} \end{aligned}$$

Nos valores apresentados na simulação até aqui temos: $MDF_{NM_2} = 3.56$, $P_{alc_2} = 390$, $P_{lim_2} = 316,84$ e $P_{max_2} = 445$. Do que se conclui que o NM_2 estaria em situação: *Suficiente*.

Agora pode-se processar todos os dados da tabela 18 do Apêndice C e determinar os dados resumidos com os valores indicados para a escala de pontuação (P_{alc_i} , P_{lim_i} e P_{max_i}) e para P_{lim_i} . Desta forma tem-se a tabela 15, Quadro Geral dos Níveis de Maturidade.

Tabela 15 – Quadro Geral dos Níveis de Maturidade

NM	Σ Questões	MDF	P_{min}	P_{lim}	P_{alc}	P_{max}	SfH1
NM_1	33	3.27	0	353.45	540	540	<i>Completo</i>
NM_2	25	3.56	0	316.84	390	445	<i>Suficiente</i>
NM_3	40	2.83	0	319.23	324	565	<i>Suficiente</i>
NM_4	30	4.50	0	607.50	516	675	<i>Insuficiente</i>
NM_5	38	3.34	0	424.45	525	635	<i>Suficiente</i>

Analisando os dados da tabela 15 é possível definir a última regra II para determinação do NM_i : um NM_i será *Suficiente* se, e somente se, $P_{alc_i} \geq P_{lim_i}$ e TODOS os NM_t , com $t < i$, forem *Completo*.

Desta forma, aplicando a regra II para os dados da tabela 15 verifica-se que o NM_3 passa a ser *Insuficiente*, uma vez que o NM_2 não é *Completo*. Desta forma a tabela 15 deve ser reescrita na forma apresentada na tabela 16

Tabela 16 – Quadro Geral dos Níveis de Maturidade - Revisado

NM	Σ Questões	MDF	P_{min}	P_{lim}	P_{alc}	P_{max}	SfH1
NM_1	33	3.27	0	353.45	540	540	<i>Completo</i>
NM_2	25	3.56	0	316.84	390	445	<i>Suficiente</i>
NM_3	40	2.83	0	319.23	324	565	<i>Insuficiente</i>
NM_4	30	4.50	0	607.50	516	675	<i>Insuficiente</i>
NM_5	38	3.34	0	424.45	525	635	<i>Insuficiente</i>

Pode-se, agora, fazer algumas observações sobre as melhorias que devem ser apresentadas para os dados analisados. Na tabela 16 nota-se que $P_{alc_2} < P_{max_2}$, embora NM_2 seja *Suficiente*. Isso é reflexo da existência de questões do NM_2 que foram respondidas com valor inferior à 5. Existem, então, artefatos sem o domínio pleno de uso. Portanto existem melhorias associadas à estas questões que podem ser recomendadas para a *software house* simulada nos dados. Do mesmo modo, para todos os NM_i onde $P_{alc_i} < P_{max_i}$.

Assim se encerra a apresentação dos parâmetros de cálculo do NM e demonstração de seu desenvolvimento com uso de dados simulados. Foram apresentados os conceitos, os parâmetros e regras para caracterizar como se determina o NM a partir de uma avaliação.

5 CONSIDERAÇÕES FINAIS

A Tecnologia foi usada para auxiliar a organização e implementação de produtos e serviços que desenvolvam a maturidade no gerenciamento de projetos. O conceito de maturidade em gerenciamento de projetos foi apresentado e estruturado. Nível de Maturidade foi associado aos conceitos de áreas de gerenciamento nas metodologias de gerenciamento de projetos. O Nível de Maturidade foi estruturado em uma escala, um modelo de associação foi estabelecido permitindo que o NM pudesse ser mensurado.

Este modelo e o uso de TIC tornou possível especificar como o conceito pode ser materializado em um meio computacional através de modelos e detalhes de procedimentos de cálculo. Vinculado ao processo de avaliação, indicou-se a estrutura para determinação de melhorias em processos de software viabilizando aumentar o Nível de Maturidade. Esse trabalho aponta, então, para a possibilidade de construir os bancos de dados e os programas; depois administrar o desenvolvimento dos dados que darão a feição final à ferramenta que permite detectar e evoluir o nível de maturidade em gerenciamento de projeto de software.

Assim conclui-se que é viável o desenvolvimento de um conjunto de programas aplicativos que alimentem um banco de dados com questões e melhorias associadas, possibilitando analisar o nível de maturidade em gerenciamento de projetos de software de *software houses* que se proponham a serem avaliadas. Quando isso ocorre as empresas podem receber um conjunto de melhorias associadas aos itens de gerenciamento que ainda faltam-lhes ser desenvolvidos.

Para possibilitar o desenvolvimento do GAIA-SANM foi apresentado a estrutura de um banco de dados, uma definição das funcionalidades esperadas dos programas e os conceitos de nível de maturidade e melhorias associadas aos itens de gerenciamento das áreas de gerenciamento de projetos de software. A partir destas conceituações apresentou-se a fundamentação do cálculo para determinação do Nível de maturidade.

A análise de dados foi executada com uma simulação onde se apresentou a estrutura de cálculo, indicando seus elementos e procedimentos, partindo de situação apresentadas nas respostas à avaliações de maturidade, processando e apresentando como resultado o nível de maturidade em desenvolvimento de software.

Na análise de dados a simulação foi apresentada em formato de tabela já resumindo o processamento das respostas às questões de uma avaliação respondida por uma *software house*. Estas tabelas podem ser introduzidas como formatos de relatórios que serão oferecidos no conjunto de serviços de *feedback* apresentados para as empresas desenvolvedoras de software.

O conjunto de programas que implementará o GAIA-SANM pode ser especificado no padrão MVC e desenvolvido com programas de código aberto. Usando para isso um ambiente de programação e desenvolvimento baseado em um servidor de páginas, um interpretador de linguagem CGI e um SGBD Relacional. Neste trabalho o banco de dados foi implementado, uma parte dos programas aplicativos desenvolvidos, faltando o desenvolvimento dos relatórios. O ambiente de desenvolvimento contou com o Servidor de página Apache, a linguagem PHP e o SGBD PostgreSQL.

Finalmente, como proposta de continuidade, faz-se a sugestão que se disponibilize um equipamento, pessoal e tempo; para que se implemente as especificações apresentadas terminando a construção do SANM, desenvolver a prospecção de *software houses* dispostas a participar das avaliações e constatar os resultados de aprimoramento no gerenciamento de projetos com o uso do SANM.

A comunidade de desenvolvedores poderá, então, referenciar o framework como um repositório de conhecimento em gerenciamento de projetos de desenvolvimento de software.

REFERÊNCIAS

- [1] DINSMORE, P. C.; CABANIS-BREWIN, J. *AMA-Manual de Gerenciamento de Projetos*. [S.l.]: Brasport, 2009.
- [2] MAGALHÃES, I.; PINHEIRO, W. *Gerenciamento de serviços de TI na prática: uma abordagem com base na ITIL : inclui ISO/IEC 20.000 e IT Flex*. Novatec, 2007. (Série Gerenciamento de TI). ISBN 9788575221068. Disponível em: <<https://books.google.com.br/books?id=zoGhqp5yu9QC>>.
- [3] VARGAS, R. V. *Gerenciamento de Projetos (8ª Edição): Estabelecendo diferenciais competitivos*. [S.l.]: Brasport, 2016.
- [4] PRESSMAN, R. S. *Engenharia de software: uma abordagem profissional*. 7ª edição. Ed: McGraw Hill, 2011.
- [5] PMI. *A Guide to the Project Management Body of Knowledge: PMBOK Guide*. Project Management Institute, 2013. (PMBOK® Guide Series). ISBN 9781935589679. Disponível em: <<https://books.google.com.br/books?id=FpatMQEACAAJ>>.
- [6] VIEIRA, M. F. *Gerenciamento de projetos de tecnologia da informação*. [S.l.]: Gulf Professional Publishing, 2007.
- [7] COBIT 5: A Business Framework for the Governance and Management of Enterprise IT. ISACA, 2012. (COBIT® 5). ISBN 9781604202373. Disponível em: <<https://books.google.com.br/books?id=1iLKVIOIg9EC>>.
- [8] HIRAMA, K. *Engenharia de Software: Qualidade e Produtividade com Tecnologia*. Elsevier Brasil, 2011. ISBN 9788535248814. Disponível em: <<https://books.google.com.br/books?id=tYhQYH2yiiYC>>.
- [9] HIRSCHFELD, H. *Planejamento com pert-cpm e análise do desempenho; métodos (manual e por computadores eletrônicos) aplicados a todos os fins: construção civil, marketing etc*. 7ª edição rev. ampl.). In: . [S.l.]: Atlas, São Paulo, 1974.

Apêndices

APÊNDICE A – DICIONÁRIO DE DADOS DO SANM

O SANM tem um banco de dados que aceita os dados das tabelas que implementam o modelo de dados indicado na Figura 4. As entidades deste modelo são convertidas em tabelas e seus arquivos construídos com o uso do SGBD Relacional PostgreSQL. Os campos das tabelas constituem parte do dicionário de dados do SANM. Os campos das tabelas implementadas neste sistema estão apresentados na Tabela 17. Todos os campos têm o preenchimento requerido (não nulos).

Tabela 17 – Dicionário de Dados do Framework SANM

Administradores		
Nome	Tipo de Dado	Comentário
cploginadmin	character varying(50)	PK da Tabela (e-mail)
txsenhaadmin	character varying(250)	Senha
txnomeadm	character varying(250)	Nome completo
aosituacao	character(1)	(A)tivo / (D)esligado / (N)ovo (aguarda analise para Ativação) / (S)uperAdm
dtcadadm	date	Data de Registro
areadegestao		
Nome	Tipo de Dado	Comentário
cpareagestao	integer	PK da Tabela
txnomearea	character varying(250)	Nome
txdescricao	character varying(250)	Descrição
dtcadareagestao	date	Data de registro
artefatos		
Nome	Tipo de Dado	Comentário
cpartefato	integer	PK da tabela
cenm	integer	FK para a tabela niveisdematuridade.
txsigla	character(6)	Sigla do artefato
txdescricao	character varying(250)	Descrição detalhada
dtcadartefato	date	Data de registro
continua...		

Tabela 17 – continuação

avaliacaoquestao		
Nome	Tipo de Dado	Comentário
ceavaliacao	integer	FK para a tabela avaliações.
cequestao	integer	FK para a tabela questões.
vlindice	integer	Valor do índice de importância atribuído à questão pelo Gestor da Desenvolvedora que responde ao Questionário de Avaliação.
avaliacoes		
Nome	Tipo de Dado	Comentário
cpavaliacao	integer	PK da Tabela
txdescritivo	character varying(500)	Descritivo da Avaliação
celogindesenv	character(10)	FK para a Desenvolvedora
cegerente	character varying(50)	FK para o Gerente
ceareagestao	integer	FK para Área de Gestão
aosituacao	character(1)	Situação da avaliação. A - Aberta (falta completar as respostas) ou E - Encerrada.
vlnivelmaturidade	integer	Índice de Maturidade da Avaliação submetida.
dttermino	date	Data de Termino de uma avaliação.
dtcadavaliacao	timestamp without time zone	Data de registro.
cidades		
Nome	Tipo de Dado	Comentário
cpcidade	integer	PK da Tabela (incrementada de 10 em 10 por uma trigger).
txnomecidade	character varying(250)	Nome da cidade.
dtcadcidade	date	Data de Cadastro da cidade.
niveisdematuridade		
Nome	Tipo de Dado	Comentário
cpnivelmaturidade	smallint	PK da tabela.
txnomenm	character varying(250)	Nome do nível de maturidade.
txdescricao	character varying(2000)	Detalha a descrição do nível de maturidade.
dtcadnivelmat	date	Data de registro.
continua...		

Tabela 17 – continuação

desenvolvedoras		
Nome	Tipo de Dado	Comentário
cplogindesenv	character(10)	PK da Tabela, deve ser atribuída pelo responsável da empresa.
txsenhadesenv	character varying(250)	Senha de Acesso ao Sistema de Nível de Maturidade.
txnomedesenv	character varying(250)	Nome da empresa desenvolvedora de software.
nucnpj	character(15)	Número do CNPJ (Cadastro Nacional de Pessoas Jurídicas).
cecidade	integer	FK para a cidade onde se localiza a desenvolvedora.
txnomeresponsavel	character varying(250)	Nome do Responsável da Desenvolvedora perante o Sistema de Nível de Maturidade.
txemailresponsavel	character varying(250)	Endereço de e-Mail do Responsável pela desenvolvedora.
txlogradouro	character varying(250)	Logradouro da sede da Desenvolvedora.
dtcaddesenvolv	date	Data de registro.
auditores		
Nome	Tipo de Dado	Comentário
cpauditor	character varying(50)	PK da tabela. Deve ser o e-mail do auditor.
txnomeauditor	character varying(250)	Nome completo do auditor.
nutelefonecontato	character varying(15)	Número de um telephone de contato com o auditor (só números).
celoginadmin	character varying(50)	FK para a Tabela administradores.
aosituacao	character(1)	B-Bloqueado, S-Suspenso ou A-Ativo.
dtcadauditor	Date	Data de registro.
continua...		

Tabela 17 – continuação

especialistas		
Nome	Tipo de Dado	Comentário
cpespecilst	character varying(50)	PK da Tabela. E-Mail do especialista informado na Inclusão.
txnomeespecilst	character varying(250)	Nome do Especialista com até 250 caracteres.
txsenhaespecilst	character(250)	Senha do Especialista com até 250 caracteres.
ceareagestao	integer	FK para a tabela areadegestao.
cecidade	integer	FK para a tabela cidades (residência).
txexpertdesde	character(7)	Texto indicando Ano/Mês desde quando atua como Especialista.
aosituacao	character(1)	Situação do Especialista: (A)tivo / (B)bloqueado (alterado pelo Adm.).
dtcadespec	date	Data de registro.
gerentes		
Nome	Tipo de Dado	Comentário
cpgerente	character varying(50)	PK da tabela - é recomendado o uso de uma conta de e-mail.
txnomegerente	character varying(250)	Nome do Gerente.
txsenhagerente	character varying(250)	Senha de acesso ao Registro do Gerente.
celogindesenv	character(10)	FK para a tabela desenvolvedoras.
emailadd	character varying(250)	Endereço de e-Mail do Gerente.
tpgerencdesen	character(7)	Tempo de Gerência na Desenvolvedora a que está vinculado.
tpgerenctotal	character(15)	Tempo total de experiência como gerente.
aosituacao	character(1)	Atributo operacional para indicar se o gerente está Bloqueado ou Desbloqueado.
dtcadgerente	character(20)	Data de registro.

continua...

Tabela 17 – continuação

questoes		
Nome	Tipo de Dado	Comentário
cpquestao	integer	Pk da Tabela.
txquestao	character varying(500)	Texto que apresenta a questão.
ceareagestao	integer	FK indicando a areadegestão.
ceespecilst	character varying(50)	FK para a tabela especialistas (autor da questão).
nupesoatrib	integer	Peso atribuído à questão. Deve variar entre 1 e 5 e ser aferido por especialistas na área de gestão que avaliem a importância do conceito ou do artefato a ser desenvolvido na questão.
ceversionamento	char	FK indicando a versão da questão
dtheadquestao	date	Data de registro.
melhoriaquestao		
Nome	Tipo de Dado	Comentário
cequestao	integer	FK indicando a questão associada para a sugestão.
vlvalencia	integer	Valor da Valência para indicar a melhoria associada à questão.
cemelhoria	integer	FK para a tabela melhorias.
dtheadmelhoriaquest	date	Data de registro.
melhorias		
Nome	Tipo de Dado	Comentário
cpmelhoria	integer	Pk da Tabela.
txsugestao	text	Texto que apresenta a sugestão.
ceareagestao	Integer	FK para a tabela areadegestao.
ceespecilst	character varying(50)	FK para a tabela especialistas (autor da melhoria).
dtheadmelhoria	Date	Data de registro.
continua...		

Tabela 17 – continuação

desenvolveagestao		
Nome	Tipo de Dado	Comentário
celogindesenv	character(10)	FK para a Tabela desenvolvedoras. Este campo será parte da PK da tabela.
ceareagestao	integer	FK para a Tabela areadegestao. Este campo será parte da PK da tabela.
cegerente	character varying(50)	FK para a Tabela gerentes.
dtcaddesenvgestao	date	Data de registro.
questoes		
Nome	Tipo de Dado	Comentário
cpquestao	integer	Pk da Tabela
txquestao	character varying(500)	Texto que apresenta a questão.
ceareagestao	integer	FK para a areadegestao.
ceartefato	smallint	FK para a tabela artefato (item de maturidade associado à questão).
ceespecilst	character varying(50)	FK para a tabela especialistas que desenvolveu a questão.
nupesoatrib	integer	Peso atribuido à questão. Deve variar entre 1 e 5
dtcadquestao	date	Data de registro.

APÊNDICE B – CASOS DE USO DO SANM

O diagrama de Caso de Uso das Desenvolvedoras é apresentado na Figura 8. A Desenvolvedora que se apresenta ao SANM pode, na primeira interface, executar seu cadastramento, solicitar o reenvio de senha ou fazer o login, neste último caso, caso consiga ser identificada pelo sistema, a desenvolvedora poderá: incluir gerentes, desbloquear gerentes, gerar uma nova avaliação (com perguntas de todas as áreas de gestão e desde que tenha gerentes indicados para responder às questões de todas as áreas de gestão); a desenvolvedora pode ainda executar relatórios de análise da evolução de seu nível de maturidade. E por último ela pode alterar seus dados cadastrais.

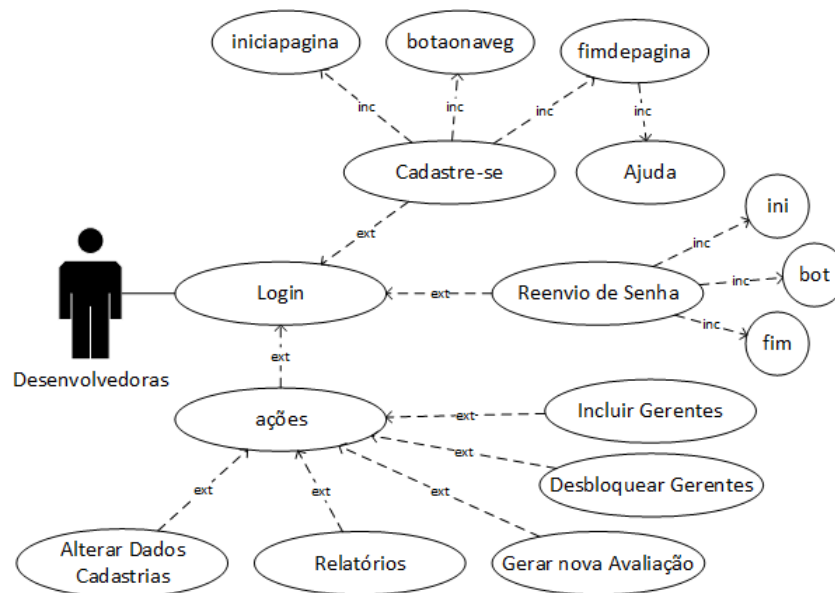


Figura 8 – Diagrama de Caso de Uso: Desenvolvedora.

Fonte: o autor

O diagrama de Caso de Uso dos Gerentes é apresentado na Figura 9. Um gerente de projeto de software é um profissional que deve estar vinculado a uma desenvolvedora. Ele pode ser gerente de uma ou mais áreas de gestão. Quando a desenvolvedora se cadastra ela pode indicar seus gerentes. Um gerente pode na tela inicial do SANM se cadastrar nos bancos de dados do sistema, nesta situação seus dados são coletados e armazenados, mas seu acesso ao sistema fica 'Bloqueado'. DEPOIS de cadastrado um gerente, o responsável pela desenvolvedora pode 'desbloquear' o acesso do gerente.

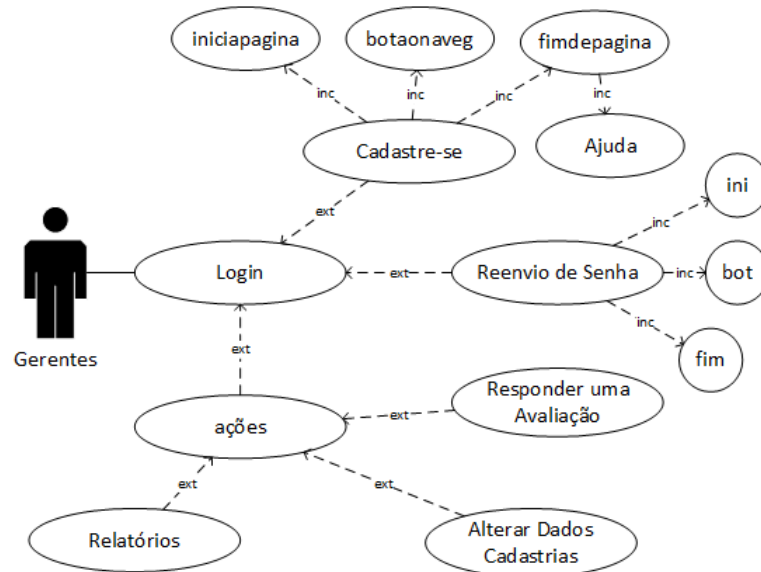


Figura 9 – Diagrama de Caso de Uso: Gerentes de Projeto de Software.

Fonte: o autor

O diagrama de Caso de Uso dos Especialistas é apresentado na Figura 10. Um especialista deve ser conhecedor profundo de uma ou mais áreas de gerenciamento e desenvolve no Gaia-SANM as questões que serão apresentadas aos gerentes nas avaliações das *software houses*. Ele também desenvolve o banco de dados das melhorias associando-as às questões das avaliações.

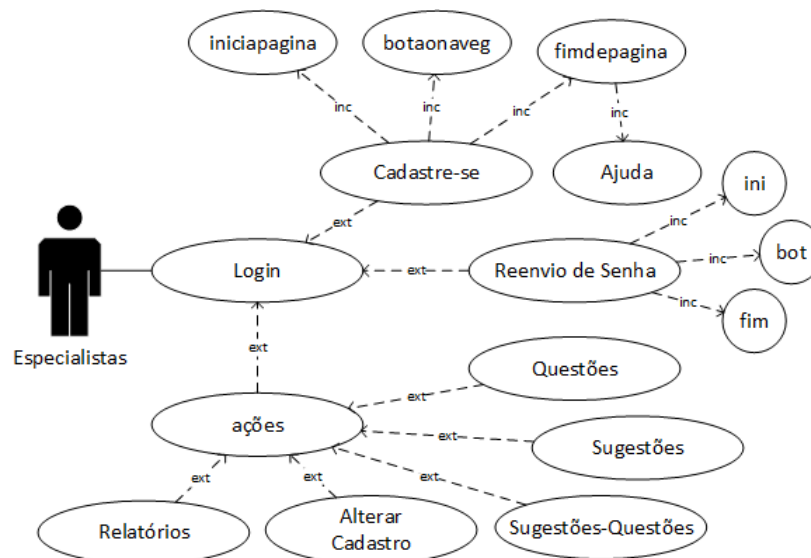


Figura 10 – Diagrama de Caso de Uso: Especialistas.

Fonte: o autor

O diagrama de Caso de Uso dos Auditores é apresentado na Figura 11. Um auditor é um papel desempenhado por um administrador do Gaia-SANM. O auditor pode inspecionar o conteúdo desenvolvido pelos especialistas podendo alterá-lo para seguir diretrizes do framework.

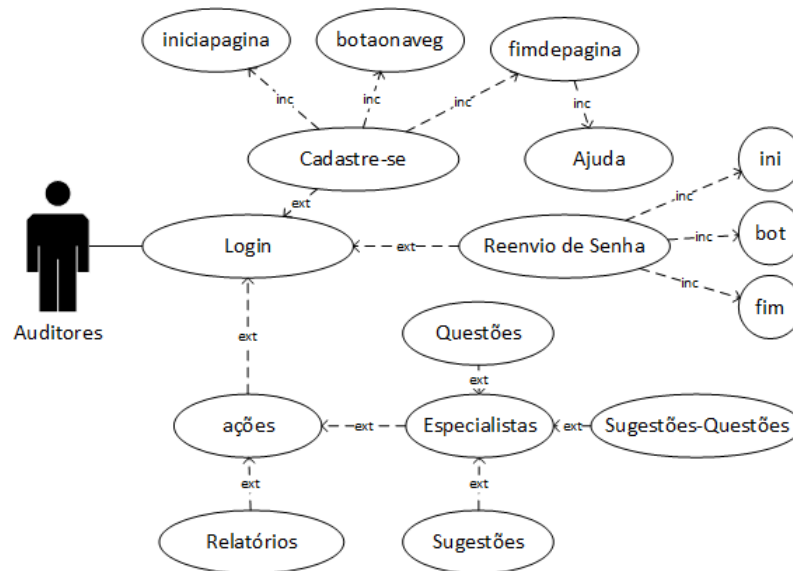


Figura 11 – Diagrama de Caso de Uso: Auditores.
Fonte: o autor

APÊNDICE C – SIMULAÇÃO DE UMA AVALIAÇÃO

Uma avaliação é gravada em algumas tabelas do banco de dados do SANM e registram as respostas que os gerentes de projeto apresentam para as questões propostas pelos especialistas em gerenciamento de projeto. As questões são agrupadas por NM. Neste apêndice apresenta-se a tabela 18 contendo os dados de uma avaliação completa.

Tabela 18 – Dados Simulados de Resposta de Avaliação para Todos os Níveis de Maturidade

NM	Questão	Peso	Resposta	Pontos
1	1	1	5	5
1	2	2	5	10
1	3	1	5	5
1	4	2	5	10
1	5	3	5	15
1	6	4	5	20
1	7	5	5	25
1	8	5	5	25
1	9	4	5	20
1	10	5	5	25
1	11	4	5	20
1	12	5	5	25
1	13	3	5	15
1	14	4	5	20
1	15	5	5	25
1	16	2	5	10
1	17	3	5	15
1	18	4	5	20
1	19	5	5	25
1	20	3	5	15
1	21	4	5	20
1	22	3	5	15
1	23	4	5	20
1	24	5	5	25
1	25	3	5	15
continua...				

Tabela 18 – continuação

NM	Questão	Peso	Resposta	Pontos
1	26	3	5	15
1	27	4	5	20
1	28	2	5	10
1	29	2	5	10
1	30	1	5	5
1	31	2	5	10
1	32	3	5	15
1	33	2	5	10
2	1	1	4	4
2	2	2	5	10
2	3	1	4	4
2	4	2	5	10
2	5	3	4	12
2	6	4	4	16
2	7	5	5	25
2	8	5	5	25
2	9	4	4	16
2	10	5	5	25
2	11	4	4	16
2	12	5	5	25
2	13	3	4	12
2	14	4	5	20
2	15	5	4	20
2	16	2	3	6
2	17	3	5	15
2	18	4	4	16
2	19	5	5	25
2	20	3	3	9
2	21	4	5	20
2	22	3	4	12
2	23	4	5	20
2	24	5	3	15
2	25	3	4	12
3	1	1	3	3
3	2	2	4	8
continua...				

Tabela 18 – continuação

NM	Questão	Peso	Resposta	Pontos
3	3	1	3	3
3	4	2	4	8
3	5	1	3	3
3	6	2	2	4
3	7	3	2	6
3	8	2	2	4
3	9	3	2	6
3	10	2	4	8
3	11	3	3	9
3	12	2	4	8
3	13	3	3	9
3	14	2	4	8
3	15	1	3	3
3	16	2	4	8
3	17	1	3	3
3	18	2	4	8
3	19	3	3	9
3	20	4	4	16
3	21	5	3	15
3	22	4	4	16
3	23	5	3	15
3	24	4	4	16
3	25	5	2	10
3	26	4	2	8
3	27	5	2	10
3	28	3	2	6
3	29	3	3	9
3	30	2	2	4
3	31	3	3	9
3	32	2	2	4
3	33	4	3	12
3	34	3	2	6
3	35	5	3	15
3	36	3	2	6
3	37	4	3	12
continua...				

Tabela 18 – continuação

NM	Questão	Peso	Resposta	Pontos
3	38	2	2	4
3	39	3	3	9
3	40	2	2	4
4	1	4	3	12
4	2	5	4	20
4	3	4	3	12
4	4	5	4	20
4	5	4	3	12
4	6	5	4	20
4	7	4	3	12
4	8	5	5	25
4	9	4	4	16
4	10	5	5	25
4	11	4	4	16
4	12	5	3	15
4	13	5	4	20
4	14	5	3	15
4	15	4	4	16
4	16	4	3	12
4	17	5	4	20
4	18	4	4	16
4	19	5	3	15
4	20	4	5	20
4	21	5	3	15
4	22	4	5	20
4	23	5	4	20
4	24	4	5	20
4	25	5	4	20
4	26	4	5	20
4	27	5	3	15
4	28	4	4	16
4	29	5	3	15
4	30	4	4	16
5	1	2	3	6
5	2	3	4	12
continua...				

Tabela 18 – continuação

NM	Questão	Peso	Resposta	Pontos
5	3	2	3	6
5	4	4	4	16
5	5	3	5	15
5	6	4	4	16
5	7	3	5	15
5	8	4	4	16
5	9	5	5	25
5	10	4	4	16
5	11	3	5	15
5	12	4	4	16
5	13	5	5	25
5	14	4	4	16
5	15	3	5	15
5	16	4	4	16
5	17	3	5	15
5	18	2	4	8
5	19	3	5	15
5	20	2	4	8
5	21	3	5	15
5	22	4	4	16
5	23	3	3	9
5	24	2	4	8
5	25	3	5	15
5	26	4	4	16
5	27	5	3	15
5	28	4	3	12
5	29	3	4	12
5	30	2	5	10
5	31	3	4	12
5	32	4	3	12
5	33	3	4	12
5	34	4	5	20
5	35	3	4	12
5	36	4	4	16
5	37	3	3	9
continua...				

Tabela 18 – continuação

NM	Questão	Peso	Resposta	Pontos
5	38	3	4	12

TRABALHOS PUBLICADOS PELO AUTOR

Trabalhos publicados pelo autor durante o programa.

1. HYPÓLITO, João M.; Barros, Rodolfo M.; **Gestão de Tempo no Projeto de Software - Bases de Cálculo do Nível de Maturidade**, 9^a Conferência Ibérica de Sistemas e Tecnologias de Informação, 8 páginas, 2014, 9^a Cisti, Barcelona, Spain.