



UNIVERSIDADE FEDERAL DE SÃO JOÃO DEL REI
CAMPUS ALTO PARAÓPEBA
ENGENHARIA DE BIOPROCESSOS

KARLA THEODORA DE C. REIS

**AVALIAÇÃO DA QUALIDADE DO GERENCIAMENTO DE PROJETO DE
ENGENHARIA DE SOFTWARE UTILIZANDO A LÓGICA FUZZY**

OURO BRANCO - MG
2022

KARLA THEODORA DE C. REIS

**AVALIAÇÃO DA QUALIDADE DO GERENCIAMENTO DE PROJETO DE
ENGENHARIA DE SOFTWARE UTILIZANDO A LÓGICA FUZZY**

Trabalho de Conclusão de Curso apresentado ao
Curso Engenharia de Bioprocessos da
Universidade Federal de São João del-Rei,
Campus Alto Paraopeba, como parte das
exigências para a obtenção do título de Bacharel
em Engenharia de Bioprocessos.

Orientador: Prof. Dr. Edson Romano Nucci

OURO BRANCO - MG
2022

AGRADECIMENTOS

O TRABALHO DESENVOLVIDO SEGUE AS NORMAS DA REVISTA
BRAZILIAN JOURNAL OF DEVELOPMENT

AVALIAÇÃO DA QUALIDADE DO GERENCIAMENTO DE PROJETO DE ENGENHARIA DE SOFTWARE UTILIZANDO A LÓGICA FUZZY

ASSESSMENT OF THE QUALITY OF SOFTWARE ENGINEERING PROJECT MANAGEMENT USING LOGIC FUZZY

Karla Theodora de Carvalho Reis

Formação acadêmica mais alta: Graduanda em Engenharia de Bioprocessos, UFSJ

Endereço completo: Campus Alto Paraopeba - Rod.: MG 443, KM 7 Ouro Branco - MG 36420 000

Email: karlatcreis15@gmail.com

Edson Romano Nucci

Formação acadêmica mais alta: Doutorado em Engenharia Química, UFSCar

Instituição de atuação atual: UFSJ

Endereço completo: Campus Alto Paraopeba - Rod.: MG 443, KM 7 Ouro Branco - MG 36420 000

Email: nucci@ufsjeu.br

RESUMO

A qualidade de software refere-se às características desejadas em produtos de software, a extensão em que um produto de software possui essas características e aos processos, ferramentas e técnicas que são usadas para garantir essas características. A Norma Internacional ISO/IEC 9126, define seis características da qualidade para produtos de software. No trabalho realizado, foram considerados os critérios de confiabilidade, usabilidade e manutenibilidade. Neste trabalho foi utilizado o software *In Fuzzy* que utiliza a lógica difusa para realizar simulações que nos possibilita verificar a qualidade do gerenciamento de um projeto de engenharia de software levando em consideração as características da qualidade estudadas e seus respectivos graus de presença dos critérios, em que possibilitou a obtenção da qualidade do gerenciamento de projeto de engenharia de software classificada como muito baixa, baixa, média, alta e muito alta.

Palavras-chave:

LÓGICA FUZZY

A lógica fuzzy, também conhecida como lógica nebulosa ou difusa, foi introduzida nos meios científicos por Lofti Asker Zadeh em 1965. Atualmente é elemento fundamental em diversos sistemas, sendo considerada uma técnica de excelência no universo computacional.

A principal motivação da teoria dos conjuntos fuzzy é o desejo de construir uma estrutura formal quantitativa, capaz de capturar as imprecisões do conhecimento humano, isto é, como esse conhecimento é formulado na linguagem atual. Essa teoria visa ser a ponte, que une modelos matemáticos tradicionais, precisos, de sistemas físicos, e a representação mental, geralmente imprecisa, desses sistemas (DUBOIS, 1991).

A teoria trata dos conjuntos não totalmente verdadeiros nem tampouco dos totalmente falsos (COSENZA et. al., 2006). Ela surge através de sua formulação da teoria dos conjuntos para gerar uma certa flexibilização da rigidez numérica da matemática clássica, o que promove mais praticidade para modelamentos matemáticos. Essa teoria é baseada no fato de que os conjuntos existentes no mundo real não possuem limites precisos (SIMÕES e SHAW, 2007). O conceito fuzzy se baseia em incertezas, ou seja, é usado para situações em que não é possível responder simplesmente "sim" ou "não". Mesmo conhecendo as informações necessárias sobre a situação, dizer algo entre "sim" e "não", como "talvez" ou "quase", torna-se mais apropriado (LUCAS, 2011).

Com estes sistemas pode-se realizar “operações com palavras”, onde os conjuntos fuzzy são os “valores” das palavras. O emprego destas para expressar as idéias forma uma linguagem. Um idioma é um modelo de expressão dos pensamentos, através de palavras. A teoria dos conjuntos fuzzy é uma linguagem que, por sua forma de expressão, forma um idioma (BRAGA et al., 1995). Os conjuntos fuzzy podem ser vistos como uma generalização da noção de conjunto na qual a função de pertinência assume valores no intervalo $[0,1]$ (FARIA, et. al., 2008).

Segundo Belchior (1997) um número fuzzy deve capturar a concepção intuitiva de números ou intervalos aproximados, tal como “valores que estão próximo de certo número real”, ou “valores que estão em torno de um dado intervalo de números reais”. Tais conceitos são essenciais para a caracterização dos estados das variáveis fuzzy e, conseqüentemente, é importante para aplicações tais como controle fuzzy, tomada de decisão, raciocínio aproximado e estatística. Atualmente diversos problemas precisam ser resolvidos a partir de sistemas complexos, entre eles está a análise da qualidade do gerenciamento de projetos de engenharia de software. Dessa maneira, com a Lógica Fuzzy é possível a construção de várias regras que facilitam visualizar o problema, fazendo com que eles se tornem menos complexos.

QUALIDADE DE SOFTWARE

Cada vez mais os produtos com características que satisfaçam as expectativas e necessidades dos usuários tornam crescente a motivação da análise da qualidade de software que atendam às exigências dos clientes.

À medida que as necessidades e expectativas dos clientes estão mudando por consequência das pressões competitivas e dos avanços tecnológicos, as organizações são introduzidas a melhorar continuamente seus produtos e processos. (NBR ISO-9000)

O conceito da qualidade de software deve estar em conformidade com os requisitos do cliente, bem como antecipar e satisfazer os desejos do cliente.

De um modo geral, pode ser também considerado o grau em que o conjunto de características inerentes a um produto, processo ou sistema, cumpre com os requisitos inicialmente estipulados para estes. De acordo com a norma brasileira NBR ISO-9000, qualidade é o grau no qual um conjunto satisfaz a requisitos. Em software, os requisitos são todas as necessidades ou expectativas expressas a serem atendidas pelo software, tanto na visão dos usuários quanto dos processos de software.

A qualidade está sempre em equilíbrio entre três fatores, a qualidade intrínseca do produto, o custo e o atendimento. A qualidade intrínseca do produto, pode ser certificada pela conformidade a normas ou avaliada pela presença ou ausência de requisitos específicos. O custo pode ser compreendido como o que preço justo que o cliente está disposto a pagar. O atendimento trata-se da quantidade certa, na hora e local certo.

Desta forma, podemos afirmar que a qualidade de software é a totalidade das características de um software que lhe confere a capacidade de satisfazer as necessidades explícitas e implícitas. Sendo explícitos, todos os requisitos que o usuário consegue avaliar em relação ao software e implícito, todos os requisitos a serem atendidos no processo de desenvolvimento que normalmente são avaliados pelos desenvolvedores.

Logo, a qualidade de software refere-se às características desejadas em produtos de software, a extensão em que um produto de software possui essas características e aos processos, ferramentas e técnicas que são usadas para garantir essas características.

PROJETO DE SOFTWARE

Um software é construído seguindo os seguintes passos: Requisitos, projeto, processos e produto. Os requisitos são derivados dos processos das áreas de negócio. É fundamental que os processos a serem informatizados sejam atuais e que detenham as melhores práticas para aquele negócio, visto que as melhores práticas nos processos são essenciais para a qualidade e sucesso de um software.

Um projeto de software é o processo pelo qual os requisitos do software são traduzidos para uma representação do software que permite o seu desenvolvimento.

As etapas do projeto são o projeto preliminar e o projeto detalhado. O projeto preliminar preocupa-se com a transformação dos requisitos do software em uma arquitetura de software e de dados. Já o projeto detalhado concentra-se nos aprimoramentos da representação

arquitetural que levam à estrutura de dados detalhada e às representações algorítmicas do software.

Um projeto de Software é importante, pois fomenta a qualidade durante o processo de desenvolvimento, fornece representações do software que serão utilizadas durante o seu desenvolvimento e traduz com precisão os requisitos do cliente que deverão estar presentes no software finalizado. As características do projeto de Software se dão por ser um esforço finito, que no seu término se projeta a entrega, geração ou finalização de determinado produto. É um esforço que pode ser subdividido em unidades de trabalho que ocorrem em uma sequência predeterminada. O objetivo, a alocação de recursos e o progresso podem ser monitorados e avaliados.

De modo geral, o projeto de software é um esforço no sentido de construir um produto, dentro de determinadas especificações, que atenda às necessidades dos usuários para que executem processos operacionais e gerenciais de negócios.

GESTÃO DO PROJETO DE SOFTWARE

O desenvolvimento de software requer o uso de práticas e atividades de gerência de projetos, por exemplo, para negociação de contratos com clientes com definição de prazos, valores, cronogramas e outros. Um projeto é a soma dos objetivos, atividades, prazos, recursos necessários, riscos e incertezas. E a gestão de um projeto de software pode ser dividida em três fases distintas: planejamento, controle da execução e monitoramento.

No Planejamento ocorre a elaboração do escopo preliminar do produto e das estimativas de prazos, recursos, esforço, custos e tamanho do produto. E a definição do processo de desenvolvimento a ser adotado pelo projeto e a sua organização.

O controle da execução está relacionado com o controle da alocação de recursos, verificação e validação de produtos intermediários, controle de mudanças no escopo, refinamento e replanejamento quando necessário e acompanhamento das tarefas e orçamento conforme cronograma.

Já o monitoramento se refere a verificação do progresso do projeto, da avaliação da qualidade dos produtos intermediários, produtividade da equipe e do projeto em termos financeiros.

PROCESSO DE SOFTWARE

Um processo de desenvolvimento define quais atividades e etapas devem ser seguidas para construir e entregar um sistema de software. É um conjunto de atividades, métodos e práticas utilizados na produção e evolução do software.

Tradicionalmente, existem dois grandes tipos de processos que podem ser adotados no desenvolvimento de software, os processos em cascata (*waterfall*) e os processos ágeis (incrementais ou iterativos).

Os processos em cascata propõem que a construção de um sistema deve ser feita em etapas sequenciais, como em uma cascata de água, na qual a água vai escorrendo de um nível para o outro. As etapas são as seguintes: levantamento de requisitos, análise, projeto detalhado, codificação e testes. Finalizada essa sequência, o software é posto em produção e liberado para uso efetivo pelos seus usuários.

A ideia de processos ágeis é construir o sistema de forma incremental e iterativa. Pequenos incrementos de funcionalidade são produzidos, em intervalos de tempo menores e, logo em seguida, validados e já podem ser testados pelos usuários. Uma vez que o incremento produzido seja aprovado, o ciclo se repete.

Os processos ágeis possuem algumas características como menor ênfase em documentação, ou seja, apenas o essencial deve ser documentado, menor ênfase em planos detalhados, pois muitas vezes nem o cliente, nem os engenheiros de software têm, no início de um projeto, uma ideia clara dos requisitos que devem ser implementados. Possuem também a Inexistência de uma fase dedicada ao design, pois este também é incremental e evolui à medida que o sistema vai nascendo, ao final de cada iteração. E desenvolvimento em times pequenos, com pouco desenvolvedores. Os Processos compreendem políticas de desenvolvimento, procedimentos para o desenvolvimento, diversas técnicas e padrões para a construção de produtos e padrões de apresentação de produtos intermediários.

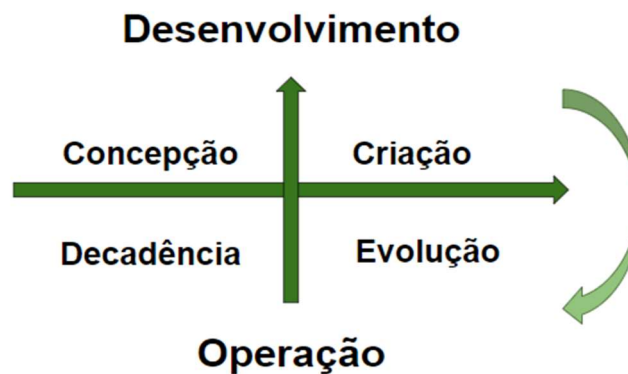
De modo geral, o processo define a forma como o projeto é executado e, consequentemente, gerenciado. Os processos de desenvolvimento de software também são muito conhecidos por metodologia de desenvolvimento.

PRODUTO DE SOFTWARE

O produto de software é o resultado da execução de um processo que contém uma série de atributos derivados dos requisitos e especificações previstos no projeto.

Como citado anteriormente a construção de um produto de software é composta pelos requisitos, projeto, processo e produto. O produto de Software é um conjunto de programas de computador, procedimentos, podendo incluir documentação e dados associados. Todo produto de software possui o ciclo de vida que pode ser demonstrado pela Figura 1

Figura 1: Ciclo de vida de um produto de software.



Fonte: Autora, 2022

O software sofre alterações durante o seu ciclo de vida caracterizadas por manutenções corretivas, manutenções preventivas, manutenções adaptativas, manutenções evolutivas e *refactoring*.

As manutenções corretivas têm como objetivo corrigir *bugs* reportados por usuários ou outros desenvolvedores. As manutenções preventivas têm como objetivo corrigir bugs latentes no código, que foram percebidos, mas ainda não causaram falhas junto aos usuários do sistema. Já as manutenções adaptativas têm como objetivo adaptar um sistema a uma mudança em seu ambiente, incluindo tecnologia, legislação, regras de negócios ou novas demandas obrigatórias nos sistemas em uso.

As manutenções evolutivas são aquelas realizadas para incluir uma nova funcionalidade ou introduzir aperfeiçoamentos importantes em funcionalidades existentes.

E *refactorings* são modificações realizadas em um software preservando seu comportamento e visando exclusivamente a melhoria de seu código ou projeto.

CARACTERÍSTICAS DA QUALIDADE

A Norma Internacional ISO/IEC 9126, define seis características da qualidade para produtos de software: funcionalidade; confiabilidade; usabilidade; eficiência; manutenibilidade e portabilidade. Para o estudo desenvolvido foi levada em consideração três características consideradas principais, são estas: confiabilidade, usabilidade e manutenibilidade.

A confiabilidade se refere à capacidade do software em manter seu nível de desempenho, sob condições estabelecidas, por um determinado período. Propõe como subcaracterísticas: maturidade, tolerância a falhas, recuperabilidade e conformidade.

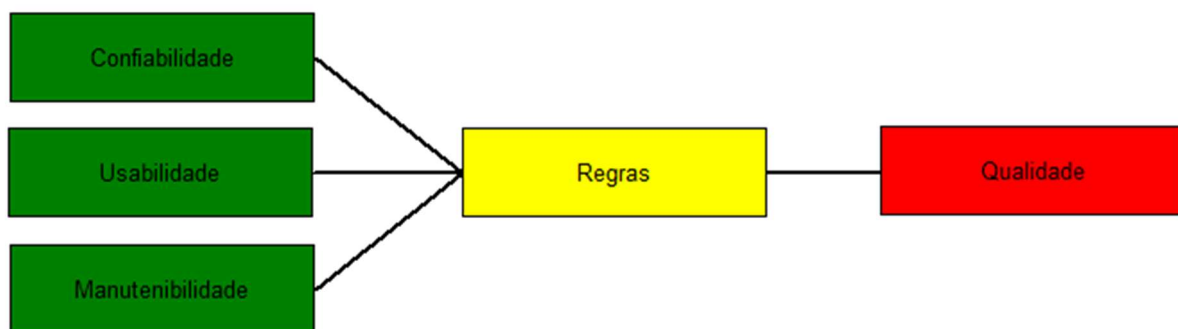
A usabilidade se refere ao esforço necessário ao uso e à homologação individual de tal uso por um conjunto de usuários estabelecido ou subentendido. Propõe como subcaracterísticas: inteligibilidade, apreensibilidade, atratividade, operatividade e conformidade.

E a manutenibilidade se refere ao esforço necessário para fazer modificações específicas no software. Propõe como subcaracterísticas: analisabilidade, modificabilidade, estabilidade, testabilidade e conformidade.

METODOLOGIA

Para a análise da avaliação da qualidade do gerenciamento de um projeto de engenharia de software foi utilizado o software InFuzzy (POSSELT *et al.*, 2015) para verificar os critérios de confiabilidade, usabilidade e manutenibilidade relacionados aos graus de presença dos critérios no projeto de forma a obter a qualidade do projeto. Na simulação realizada foram utilizados três variáveis de entrada como citadas anteriormente: confiabilidade, usabilidade e manutenibilidade. E para função de pertinência de saída: Qualidade, definida como muito baixa, baixa, média, alta e muito alta.

Figura 2: Esquema para análise da avaliação de qualidade do gerenciamento de projeto de engenharia de software, via Software InFuzzy.



Fonte: Autora, 2022

As funções de pertinência de confiabilidade, usabilidade e manutenibilidade foram classificadas em "Totalmente Ausente, Baixa Presença, Média Presença, Alta Presença e Total Presença". O termo linguístico "Totalmente Ausente" foi realizado a uma função de rampa para a esquerda. Os termos "Baixa Presença, Média Presença e Alta Presença" foram realizados a uma função triangular e o termo "Total Presença" foi realizado a uma função de rampa para a direita.

Quadro 1: Conjuntos triangulares e rampa fuzzy correspondentes aos graus de presença.

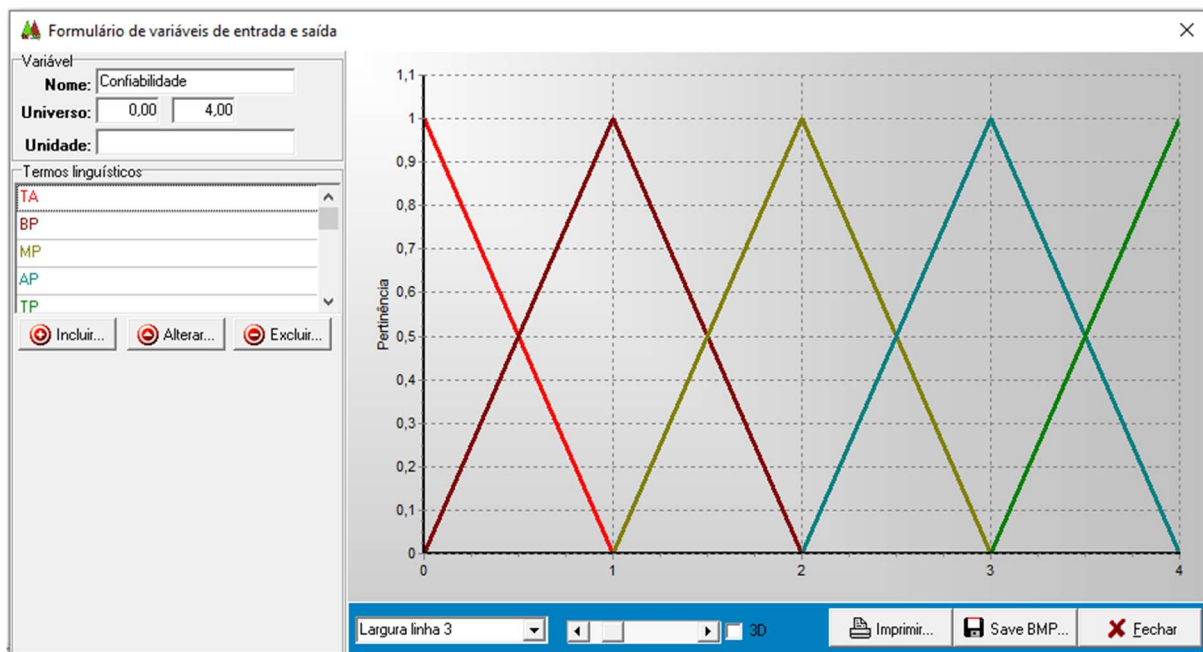
Valor do termo <i>fuzzy</i>	Número Triangular Rampa <i>fuzzy</i>	Grau de Presença
0	(0,1)	Totalmente Ausente (TA)
1	(0,1,2)	Baixa Presença (BP)
2	(1,2,3)	Média Presença (MP)
3	(2,3,4)	Alta Presença (AP)
4	(3,4)	Total Presença (TP)

Fonte: Autora, 2022

RESULTADOS E DISCUSSÃO

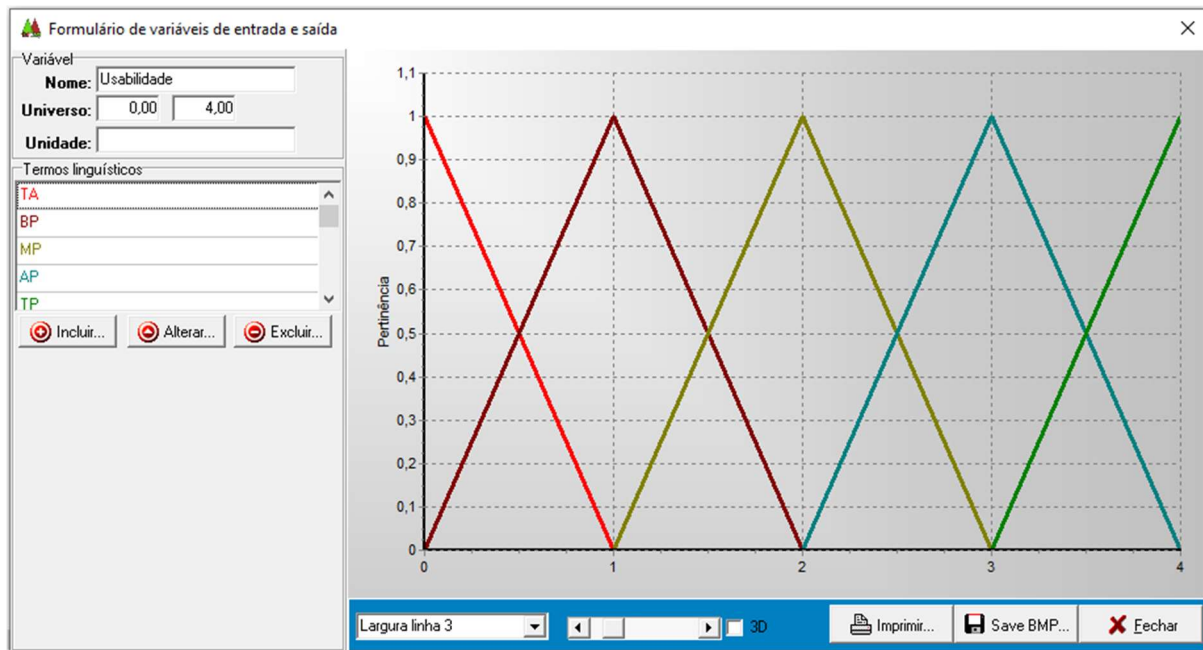
Nas Figuras 3 a 5 estão ilustradas graficamente as funções de pertinência, confiabilidade, usabilidade e manutenibilidade como citadas anteriormente.

Figura 3. Função de Pertinência: Confiabilidade



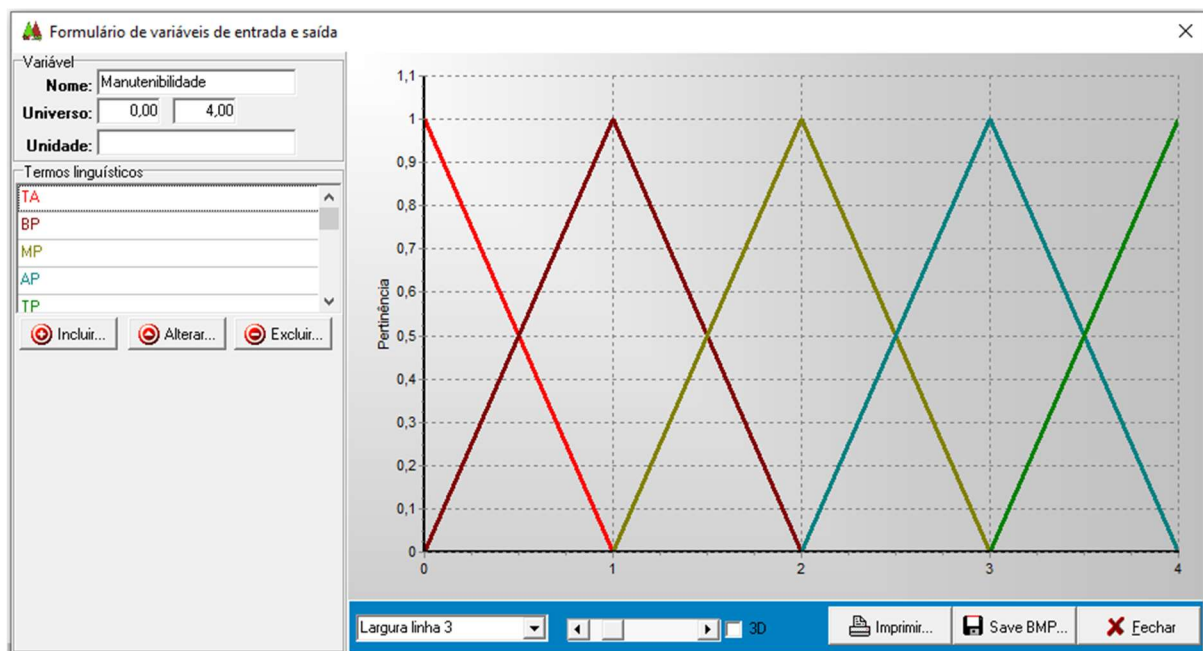
Fonte: Autora, 2022

Figura 4. Função de Pertinência: Usabilidade



Fonte: Autora, 2022

Figura 5. Função de Pertinência: Manutenibilidade



Fonte: Autora, 2022

O formato utilizado com três variáveis de entrada e cinco termos linguísticos para cada variável nos possibilita a criação de 243, mas para o sistema em estudo foram consideradas 17 principais regras para obter resultados objetivos e assertivos relacionados a qualidade, classificando-a como: muito baixa, baixa, média, alta e muito alta. A criação das regras está relacionada com a análise do esquema elaborado, e estão apresentadas na Figura 6:

Figura 6. Conjuntos de regras do sistema

Nro	Se	Antecedentes	Então	Consequentes
1	Se	(Confiabilidade = TA) AND (Usabilidade = BP) AND (Manutenibilidade = BP)	Então	(Qualidade = MB)
2	Se	(Confiabilidade = BP) AND (Usabilidade = BP) AND (Manutenibilidade = BP)	Então	(Qualidade = MB)
3	Se	(Confiabilidade = MP) AND (Usabilidade = BP) AND (Manutenibilidade = BP)	Então	(Qualidade = B)
4	Se	(Confiabilidade = BP) AND (Usabilidade = MP) AND (Manutenibilidade = MP)	Então	(Qualidade = B)
5	Se	(Confiabilidade = MP) AND (Usabilidade = MP) AND (Manutenibilidade = MP)	Então	(Qualidade = M)
6	Se	(Confiabilidade = MP) AND (Usabilidade = MP) AND (Manutenibilidade = AP)	Então	(Qualidade = M)
7	Se	(Confiabilidade = AP) AND (Usabilidade = AP) AND (Manutenibilidade = MP)	Então	(Qualidade = A)
8	Se	(Confiabilidade = AP) AND (Usabilidade = AP) AND (Manutenibilidade = AP)	Então	(Qualidade = A)
9	Se	(Confiabilidade = TP) AND (Usabilidade = TP) AND (Manutenibilidade = AP)	Então	(Qualidade = MA)
10	Se	(Confiabilidade = TP) AND (Usabilidade = TP) AND (Manutenibilidade = TP)	Então	(Qualidade = MA)
11	Se	(Confiabilidade = TA) AND (Usabilidade = BP) AND (Manutenibilidade = MP)	Então	(Qualidade = MB)
12	Se	(Confiabilidade = TA) AND (Usabilidade = MP) AND (Manutenibilidade = AP)	Então	(Qualidade = B)
13	Se	(Confiabilidade = TA) AND (Usabilidade = AP) AND (Manutenibilidade = TP)	Então	(Qualidade = M)
14	Se	(Confiabilidade = BP) AND (Usabilidade = MP) AND (Manutenibilidade = AP)	Então	(Qualidade = M)
15	Se	(Confiabilidade = BP) AND (Usabilidade = AP) AND (Manutenibilidade = TP)	Então	(Qualidade = A)
16	Se	(Confiabilidade = MP) AND (Usabilidade = AP) AND (Manutenibilidade = TP)	Então	(Qualidade = A)
17	Se	(Confiabilidade = TA) AND (Usabilidade = TA) AND (Manutenibilidade = TA)	Então	(Qualidade = MB)

Fonte: Autora, 2022

Como forma de análise da qualidade do gerenciamento de projetos, foram realizadas várias simulações no InFuzzy dos critérios de confiabilidade, usabilidade e manutenibilidade. Como resultados simulados foi possível identificar a qualidade do projeto através dos valores estabelecidos para os critérios utilizados.

Tempo (ms)	Confiabilidade	Usabilidade	Manutenibilidade	Qualidade
242	0,00	0,00	0,00	0,00
278	1,00	1,00	1,00	0,00
277	2,00	2,00	2,00	2,00
233	3,00	3,00	3,00	3,00
126	4,00	4,00	4,00	3,90
104	1,00	2,00	3,00	2,00
131	1,00	3,00	4,00	3,00
97	2,00	3,00	4,00	3,00

Fonte: Autora, 2022.

CONCLUSÃO

Com os resultados obtidos de oito simulações realizadas levando em consideração possíveis valores para a presença dos critérios analisados, é possível concluir que os valores

de saída, ou seja, da qualidade do gerenciamento, condiz com os valores dos critérios. Por meio da utilização da lógica Fuzzy, aplicando o software in Fuzzy, foi possível verificar que os critérios de confiabilidade, usabilidade e manutenibilidade, se avaliados com valor igual a 0, ou seja, totalmente ausente, a qualidade se dá por muito baixa.

Analisando um outro formato levando em consideração a confiabilidade com valor igual a 1, a usabilidade igual a 2 e a manutenibilidade igual a 3 a qualidade se dá por 2, que representa baixa qualidade devido a variação dos valores dos critérios.

Um outro cenário de possível análise é considerando os valores dos critérios todos altos, como 4 por exemplo, que se caracteriza como totalmente presente. Com esses valores máximos de critério, o resultado da qualidade é representado por muito alta.

Dessa forma, a análise através da lógica fuzzy nos possibilita verificar qualquer valor dado aos respectivos critérios levando em consideração ao seu grau de presença que será possível obter a qualidade do gerenciamento de projeto de engenharia de software.

REFERÊNCIAS

ABNT- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR ISO 9000/2000 - Sistema de Gestão da Qualidade: Fundamentos e Vocabulário. Rio de Janeiro, ABNT, 2004

ABNT - ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR ISO/IEC 9126-1: Engenharia de Software - Qualidade de produto - parte 1: modelo de qualidade. Rio de Janeiro, ABNT, 2003.

BELCHIOR, A.D. Um Modelo Fuzzy para Avaliação da Qualidade de Software. Tese de Doutorado, Engenharia de Sistemas e Computação, COPPE, Universidade Federal do Rio de Janeiro, UFRJ, RJ, 1997

BRAGA, M. J. F.; BARRETO, J. M.; MACHADO, M. A. S. Conceitos da Matemática Nebulosa na Análise de Risco. Rio de Janeiro: Artes & Rabiscus, 1995.

COSENZA, H.J.S.R. et al. Aplicação de Um Modelo de Hierarquização Como Instrumento para Tomada de Decisão: Caso de uma Multinacional. In: XXVI Encontro Nacional de Engenharia de Produção, ENEGEP, 2006, Fortaleza, 2006.

DUBOIS, D., PWE, H., 1991, Fuzzy sets in approximate reasoning, Part I: inference with possibility distributions, Fuzzy Sets and Systems, IFSA, Special Memorial Volume: 25 years of fuzzy sets, North-Holland - Amsterdam.

FARIA, M.N. et al. Um Sistema de Avaliação em EAD baseado em Lógica Fuzzy. Revista Eletrônica Horizonte Científico. Minas Gerais, edição 2008, dez/2008.

LUCAS, C. A. Uma Introdução á Lógica Fuzzy. Revista Eletrônica de Sistemas de Informação e de Gestão Tecnológica, v. 1, n. 1, p. 17–28, 2011.

KLIR, G. J. BO fuzzy sets and fuzzy logic: theory and applications. New Jersey: Prentice Hall PTR, 1995.

POSSELT, E. L.; FROZZA, R.; MOLZ, R. F. INFUZZY: Ferramenta para desenvolvimento de aplicações de sistemas difusos. **Revista Brasileira de Computação Aplicada**. Passo Fundo, v. 7, n. 1, p. 42-52, 2015.

SIMÕES, M.G. e SHAW, I.S. Controle e Modelagem Fuzzy. 2. ed. Revisada e Completa. São Paulo: Blucher: FAPESP, 2008.