



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

JONAS GOMES AGUIAR

**LEVANTAMENTO SOBRE O USO DE MODELOS GRÁFICOS NO
PROCESSO DO DESENVOLVIMENTO DE SOFTWARE**

CAMPINA GRANDE - PB

2022

JONAS GOMES AGUIAR

**LEVANTAMENTO SOBRE O USO DE MODELOS GRÁFICOS NO
PROCESSO DO DESENVOLVIMENTO DE SOFTWARE**

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Professor Dr. João Arthur Brunet Monteiro.

CAMPINA GRANDE - PB

2022

JONAS GOMES AGUIAR

**LEVANTAMENTO SOBRE O USO DE MODELOS GRÁFICOS NO
PROCESSO DO DESENVOLVIMENTO DE SOFTWARE**

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

BANCA EXAMINADORA:

Professor Dr. João Arthur Brunet Monteiro

Orientador – UASC/CEEI/UFCG

Professor Dr. Carlos Wilson Dantas de Almeida

Examinador – UASC/CEEI/UFCG

Professor Tiago Lima Massoni

Professor da Disciplina TCC – UASC/CEEI/UFCG

Trabalho aprovado em: 30 de Março de 2022.

CAMPINA GRANDE - PB

ABSTRACT

Graphical models of software development architectural, process and industry code decisions have become relevant tools over time. Lately, there has been a decline in the use of these traditional models such as UML or Model C4. In this article we want to understand how people who actually use the models are used, as a motivation to use them, in addition to showing the resources they can actually use, and who brings the most useful models. We will evaluate the development of data with people from the phenomenon community to understand how the whole phenomenon involved in the graphical models occurs. The final overview of this article is expected to actually be capable of industry models for a small use, and are evaluating whether these traditional models are skewed.

Keywords: Diagrams; Graphic Models; UML; C4 Model.

Levantamento sobre o uso de modelos gráficos no processo do desenvolvimento de software.

Jonas Gomes Aguiar
jonas.aguiar@ccc.ufcg.edu.br
Universidade Federal de Campina
Grande, Paraíba

João Arthur Brunet Monteiro
Orientador
joao.arthur@computacao.ufcg.edu.br
Universidade Federal de Campina
Grande, Paraíba

RESUMO

Os modelos gráficos das decisões arquitetônicas, de processo e de código na indústria do desenvolvimento de software se tornaram ferramentas relevantes ao longo do tempo. Ultimamente é sentida uma queda na utilização desses modelos tradicionais como UML ou Modelo C4. Neste artigo queremos entender como se dá a utilização de tais modelos gráficos, quem são as pessoas que realmente utilizam os modelos, a motivação por trás da utilização, além de mostrar os benefícios, desvantagem, e relevância. Iremos avaliar dados coletados com pessoas da comunidade do desenvolvimento de software para entender como se dá todo esse fenômeno envolto dos modelos gráficos. Espera-se que ao final deste artigo sejamos capazes de ter um pequeno panorama atual do uso desses modelos na indústria, e avaliar se realmente esses modelos tradicionais estão declinando.

Palavras-chaves: Diagramas, modelos gráficos, UML, Modelo C4

1. INTRODUÇÃO

A utilização das imagens pelo homem para demonstrar conceitos, pensamentos, ideias e fatos vem desde a sua origem. No cenário do desenvolvimento de software as imagens do mesmo modo ajudam a demonstrar conceitos, definições, e ordem envoltos na formação da arquitetura, design, da escrita e nos processos que o desenvolvimento de software impõe. Além do mais, o uso desses modelos é uma maneira clara e objetiva de comunicar como planeja construir um software como seu design e arquitetura inicial, como a equipe será organizada, requisitos não funcionais, ou como esse software funcionará, mostrando seus requisitos funcionais, demonstração de códigos-fonte que criará uma documentação retrospectiva gerando uma facilidade ao compartilhar esse conhecimento. Mas há problemas no uso desses modelos gráficos, infelizmente, eles não conseguem captar toda a complexidade dos sistemas, Frederick Brooks ao comentar sobre isso em seu livro Não Existe Bala de Prata disserta: "A complexidade de um software é uma propriedade essencial e não acidental. Portanto, representações de uma entidade de software que abstraem sua complexidade normalmente também abstraem

sua essência. Por três séculos, matemáticos e físicos obtiveram grandes avanços construindo modelos simplificados de um fenômeno complexo, derivando propriedades de tais modelos e verificando tais propriedades por meio de experimentos. Esse paradigma funcionou porque as complexidades ignoradas não são propriedades essenciais do fenômeno sob estudo. Porém, essa abordagem não funciona quando as complexidades são essenciais."[1]. E segundo Marco Tulio Valente em seu livro Engenharia de Software Moderna afirma o seguinte: "Se pensarmos em termos de atividades de desenvolvimento de software, a criação de modelos é considerada uma atividade de projeto (design). Durante o levantamento de requisitos, as atenções estão voltadas para a definição do problema que será resolvido pelo sistema. Quando se avança para atividades de projeto, o problema já deve estar devidamente entendido e as atenções se voltam para a concepção de uma solução capaz de resolvê-lo. Após essa solução ser projetada, ela deve ser implementada, usando-se linguagens de programação, bibliotecas, frameworks, bancos de dados, etc."[2] Demonstra que necessitamos avaliar a proposta do uso dos modelos, que atualmente são reduzidos ao escopo das metodologias ágeis, onde são confusos, pouco claros, e tendem a ter uma descrição pobre que leva a falta de entendimento e confusão por meio dos envolvidos no processo. Nesse artigo iremos relatar uma investigação enxuta dentro da comunidade das pessoas que estão imersas no desenvolvimento de software e tratar com tal grupo sobre questões que envolvem os modelos gráficos e sua aplicação, como o perfil de quem os utiliza e os cria, a motivação do seu uso e criação, a finalidade prática da utilização e uso, a sua frequência no desenvolvimento de software atual, a fase do desenvolvimento em que são utilizados ou criados, os benefícios e/ou desvantagens práticas e quais ainda são relevantes e/ou imprescindíveis.

2. Pesquisa e Metodologias

Este trabalho segue uma abordagem quantitativa com objetivo descritivo, usando dados coletados através de um formulário

disponível eletronicamente durante o mês de setembro do ano de 2021. Os dados foram analisados com o apoio de ferramentas de software para elucidar as questões referentes à pesquisa. Os artefatos gerados para as análises realizadas neste trabalho podem ser encontrados em uma pasta no Google Drive[3]. Nesta seção, serão detalhados os dados, as questões de pesquisa e as técnicas utilizadas neste trabalho.

2.1 Objetivos e Questões de Pesquisa

Este trabalho tem como objetivo explorar como a indústria de desenvolvimento de software tem se utilizado dos modelos gráficos através de uma análise quantitativa dos dados gerados através das respostas contidas em um formulário disponibilizado eletronicamente, considerando perfil, motivação de uso e construção, exemplos de representações, benefícios e desvantagens relacionados com a utilização dos modelos gráficos. Buscando atingir esse objetivo, foram expostas as seguintes questões de pesquisa:

- Q1: Qual a experiência em relação ao desenvolvimento de software?
- Q2: Qual o cargo atual?
- Q3: Para o que especificamente utilizou/utiliza modelos gráficos dentro do desenvolvimento de software?
- Q4: Já fez/faz ou apenas utiliza modelos gráficos criados por terceiros em seu ambiente de trabalho?
- Q5: Por qual motivo se utiliza de modelos gráficos?
- Q6: Concorda que o uso de modelos gráficos ajudam na compreensão de processos, design e arquitetura do software.
- Q7: Os modelos gráficos que já utilizou/utiliza dentro do ambiente de trabalho do desenvolvimento de software.
- Q8: Quais modelos gráficos relevantes dentro do desenvolvimento de software?
- Q9: Modelos gráficos são imprescindíveis na documentação de um software.
- Q10: Quais modelos gráficos são imprescindíveis dentro do desenvolvimento de software?
- Q11: Os benefícios do uso de modelos gráficos.
- Q12: Quais são as desvantagens do uso de modelos gráficos dentro do desenvolvimento de software?
- Q13: Toda documentação de software deve conter modelos gráficos.

2.2 Dados do estudo

Foram utilizados dados recolhidos em um formulário, disponibilizado no formato .xlsx. O formulário reúne dados sobre: perfil do contribuinte da pesquisa, utilização de modelos, criação de modelos, tipos de modelos, relevância e opinião sobre os modelos. A pesquisa foi desenvolvida para a comunidade de desenvolvedores e compartilhada nas redes sociais como LinkedIn, Twitter e Facebook. Todos esses formulários foram realizados via Google Forms[3] e podem as respostas serem acessadas publicamente[4][5]. Ao todo foram coletadas 136 respostas.

2.3 Procedimentos

Para a realização das análises ambicionadas neste trabalho, foi essencial a adoção de um processo envolvendo duas etapas: tratamento e processamento.

Tratamento: Nesse passo foi realizado um tratamento dos dados, visando suprimir ruídos, padronizar respostas e preencher dados faltantes respeitando o contexto dos dados presentes. Como o conjunto de dados foi pequeno, o tratamento foi realizado manualmente através da própria ferramenta de armazenamento (Google Spreadsheets). Os tratamentos feitos foram:

- Correção de ortografia das respostas;
- Padronizar respostas que tinham a mesma semântica mas escrita distinta;
- Remover inconsistências devido ao mau preenchimento dos formulários (exemplo: marcar Outro e não preencher o que seria esse outro);

Processamento: Aqui os dados foram pré-processados utilizados em análise exploratória para desta forma conseguir um entendimento básico dos dados e das relações existentes entre as variáveis analisadas, visando encontrar as principais métricas e estatísticas deles. Foram utilizadas métricas conhecidas da estatística, tais como frequências (absolutas e relativas). A mesma ferramenta utilizada na fase de pré-processamento, também foi utilizada nessa fase para a construção dos gráficos. É importante salientar que nesse passo fizemos a maioria da análise baseada em três grupos de respondentes: todos os participantes, participantes com até cinco anos de experiência no mercado de trabalho e participantes com mais de cinco anos de experiência no mercado de trabalho. Isso foi feito para investigarmos como se comportam os profissionais menos e mais experientes de acordo com a temática, e podermos traçar um diagnóstico melhor. Em alguns tópicos específicos foi mais relevante para a pesquisa analisar apenas o grupo geral.

3. RESULTADOS E DISCUSSÕES

Nas subseções seguintes, serão apresentados os resultados obtidos na análise com base nas questões de pesquisa propostas.

3.1 Perfil dos respondentes



Figura 1: Experiência profissional dos respondentes

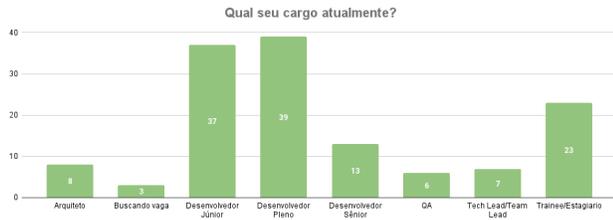


Figura 2: Principais cargos dos respondentes

O primeiro passo para analisar os dados é entender o público que participou da pesquisa. Ao todo tivemos um número considerável de respondentes no total de 136 pessoas. Dentre elas, 22,8% tem mais do que cinco anos de experiência na área, e 58% possui mais de um ano de experiência e menos do que cinco anos. Isso demonstra que o nosso público possui um número consideravelmente importante de 80,8% de pessoas que estão há no mínimo um ano no mercado de trabalho.

Outro dado importante é o número de cargo dos respondentes, com 63 respondentes(43,32%) sendo Desenvolvedores Júnior[6], Trainee/Estagiário ou buscando vaga. E o restante 53,67% se dividindo entre Desenvolvedores Pleno, Sênior, Tech/Team Lead[7], Arquitetos[8] e QA(Quality Assurance)[9]. Essa quase igualdade de cargo que requer experiência técnica maior e outros que não requerem tanta experiência técnica nos ajudam a traçar uma pesquisa mais igualitária.

3.2 Criação e Utilização de Modelos Gráficos

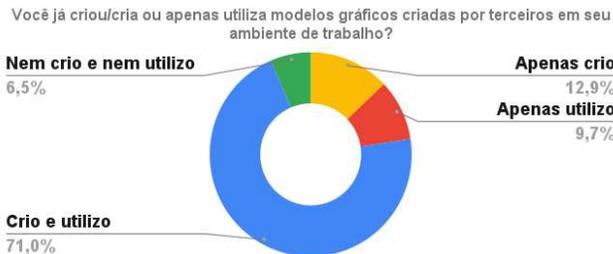


Figura 3: Frequência absoluta da criação ou utilização de Modelos Gráficos no público total



Figura 4: Frequência relativa da criação ou utilização de Modelos Gráficos em público com cinco anos ou menos de experiência.

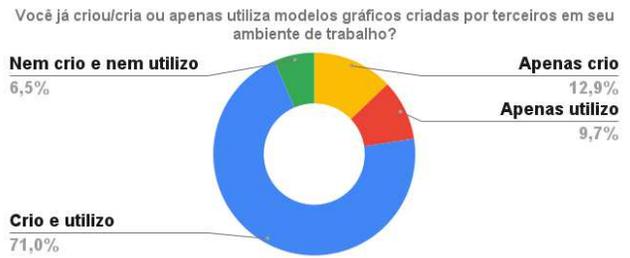


Figura 5: Frequência relativa da criação ou utilização de Modelos Gráficos em público com mais de cinco anos de experiência.

Podemos notar nesses gráficos acima a distribuição do uso ou criação de modelos gráficos pelos participantes da pesquisa. 81,5% dos participantes já utilizaram modelos gráficos em seu trabalho, 84,7% dos participantes já criaram algum tipo de modelo gráfico e apenas 6,5% não criam e nem utilizam.

Um fato destacável no comparativo das figuras 4 e 5 são as diferenças nos números de pessoas que apenas utilizam os modelos, que é de 26,6% no público com menos de 5 anos de experiência e 9,7% no público com mais de 5 anos de experiência. Também outra diferença a se notar é o do item “apenas criou” que é respectivamente 1,0% e 12,9%.

3.3 Objetivo do Uso de Modelos Gráficos

Para o que especificamente você cria/criou modelos gráficos dentro do desenvolvimento de software?

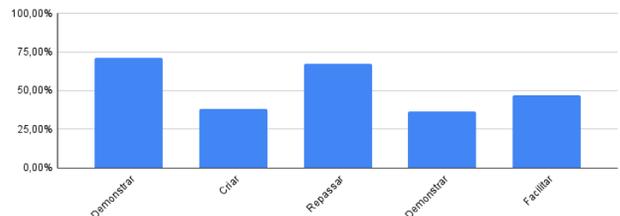


Figura 6: Para que o público geral cria ou criou modelos gráficos

Para o que especificamente você cria/criou modelos gráficos dentro do desenvolvimento de software?

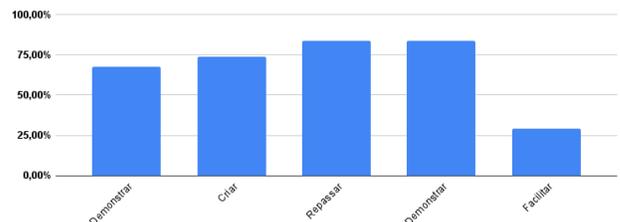


Figura 7: Para que o público com mais de 5 anos de experiência cria ou criou modelos gráficos

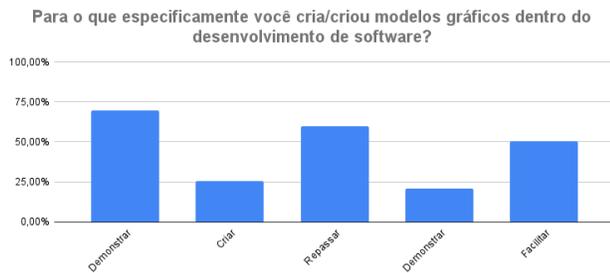


Figura 8: Para que o público com 5 anos ou menos de experiência cria ou criou modelos gráficos

Nessa subseção trataremos da finalidade no uso dos modelos gráficos. As respostas para essa questão se dividiam em cinco: Demonstrar fluxos do funcionamento, Criar Documentação, Repassar conhecimento para membros do seu projeto, Demonstrar arquitetura geral e específica do software aos clientes e/ou desenvolvedores e Facilitar compreensão das demandas diárias do desenvolvimento. Vemos que o público geral dos respondentes tem como finalidade principal a demonstração de fluxos(71,2%), em segundo lugar é repassar conhecimento para os membros da equipe(67,4%) e em terceiro para facilitar na compreensão das demandas diárias do desenvolvimento(47%). Criar documentação e demonstrar arquitetura geral e específica do software para clientes aos clientes e/ou desenvolvedores tiveram citação de 37,9% e 36,4%, respectivamente.

No comparativo entre os dois grupos(figuras 7 e 8) vale destacar a diferença significativa do público com cinco anos ou mais de experiência nos quesitos de criação de documentação e demonstração arquitetura geral e específica do software para clientes aos clientes e/ou desenvolvedores com uma diferença de 74,19% a 25,71% e de 83,87% a 20,95% respectivamente. O único quesito que se manteve praticamente igual em ambos os grupos foi o de demonstrar fluxos do funcionamento que ficou em 69,52% e 67,54%.

3.4 Motivação Para o Uso de Modelos Gráficos

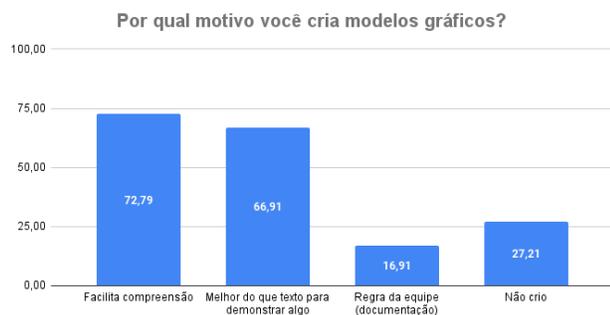


Figura 9: Motivação por trás do uso de modelos gráficos pelo público geral

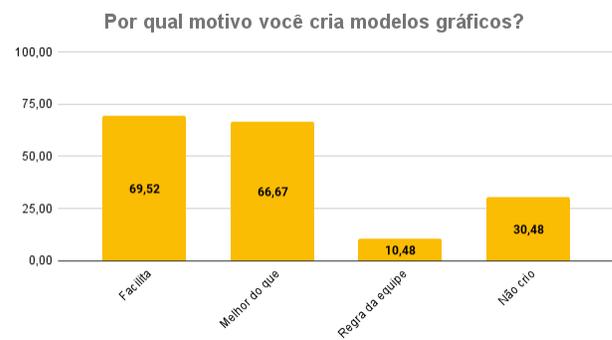


Figura 10: Motivação por trás do uso de modelos gráficos pelo público com 5 anos ou menos de experiência.

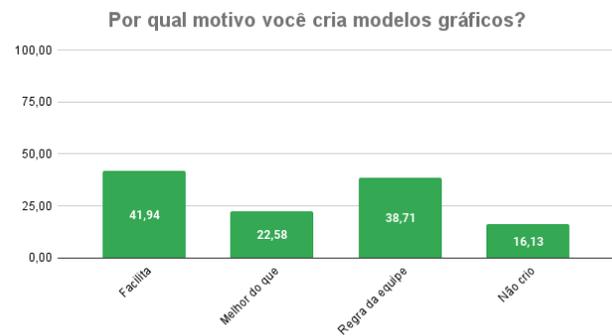


Figura 11: Motivação por trás do uso de modelos gráficos pelo público com mais de 5 anos de experiência.

A motivação por trás do uso de modelos são principalmente que eles facilitam a compreensão(72,79%) e que é melhor do que texto para demonstrar algo(66,91%) sendo essa a opinião do público geral, onde as possíveis respostas poderiam ser:

Facilita compreensão, Melhor do que texto para demonstrar algo, Regra da equipe(documentação) e Não crio. Regra da equipe e a não criam tiveram a citação de 16,71% e 27,21%, respectivamente.

No comparativo entre os grupos vale notar que todos os quesitos tiveram diferenças de um grupo para o outro. Mas Regra da equipe e Não Crio tiveram as maiores diferenças de 10,48% a 38,71% e 30,48% a 16,13% do grupo com menos de 5 anos de experiência para o grupo com mais de 5 anos de experiência.

3.5 Benefícios e Desvantagens do uso de Modelos Gráficos

4.5.1 Benefícios

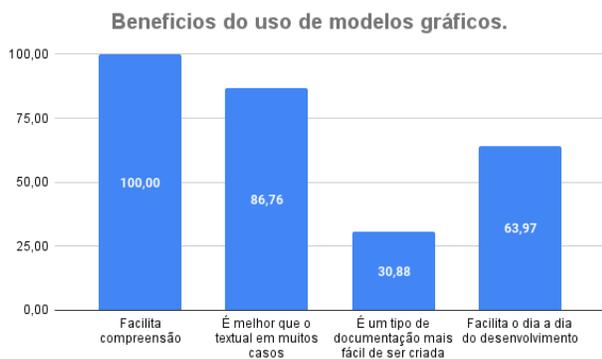


Figura 11: Benefícios do uso de modelos gráficos pelo público geral.

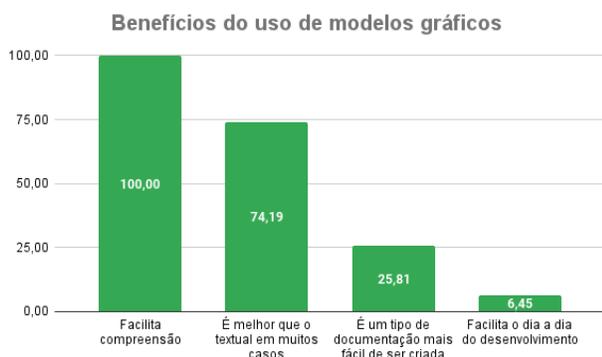


Figura 12: Benefícios do uso de modelos gráficos pelo público com mais de 5 anos de experiência.

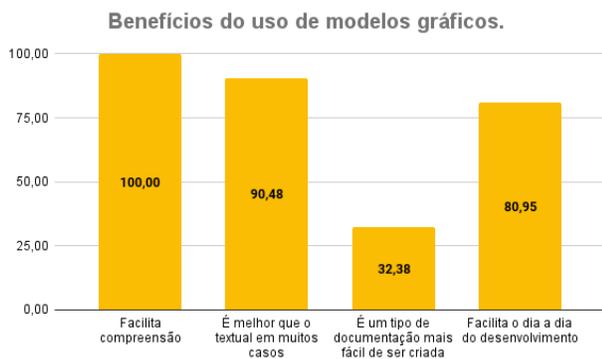


Figura 13: Benefícios do uso de modelos gráficos pelo público com 5 anos ou menos de experiência.

Sendo as opções para resposta para os benefícios do uso de modelos gráficos: Facilita compreensão, é melhor que o textual em muitos casos, é um tipo de documentação mais fácil de ser criada e facilita o dia a dia do desenvolvimento. Os principais benefícios levantados pelos respondentes foram que usar modelos gráficos facilitam a compreensão(100%), que é melhor que o textual em muitos casos(86,76%) e que facilita o dia a dia do desenvolvimento(63,97%).

Um ponto bastante interessante de se notar é que todos que responderam citaram que usar modelos gráficos facilitam a compreensão.

Na comparação dos grupos a facilitação do dia a dia do desenvolvimento foi o ponto de maior diferença, com o grupo com a experiência maior que 5 anos citando-o em apenas 6,45% e o grupo com menos de 5 anos de experiência citando-o em 80,95%.

4.5.2 Desvantagens

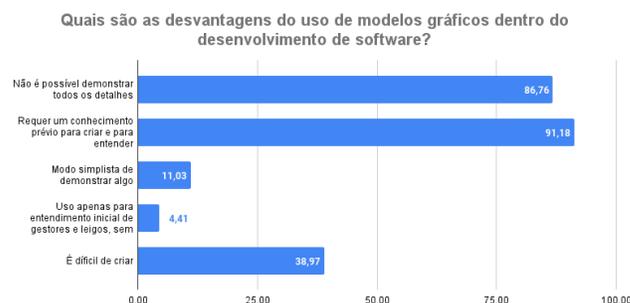


Figura 14: Desvantagens do uso de modelos gráficos pelo público geral.

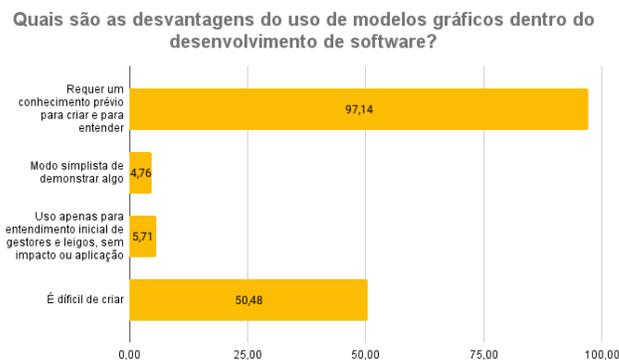


Figura 15: Desvantagens do uso de modelos gráficos pelo público com 5 anos ou menos de experiência.

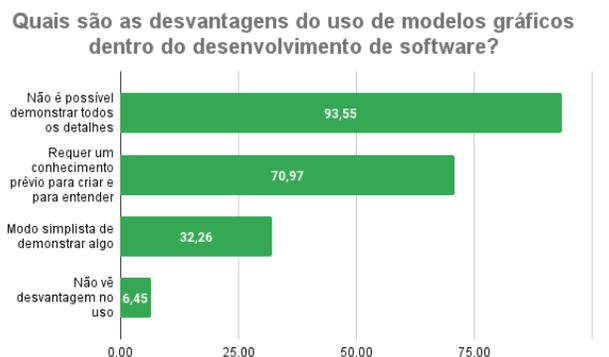


Figura 16: Desvantagens do uso de modelos gráficos pelo público com mais de 5 anos de experiência.

Sendo as principais opções das desvantagens do uso de modelos gráficos: Não é possível demonstrar todos os detalhes, Requer um conhecimento prévio para criar e para entender, Modo simplista

de demonstrar algo, Uso apenas para entendimento inicial de gestores e leigos, sem impacto ou aplicação prática no dia a dia e É difícil de criar. As principais desvantagens citadas são Requer um conhecimento prévio para criar e para entender(91,18%) e Não é possível demonstrar todos os detalhes(86,76%).

O comparativo entre os dois grupos mostra que o grupo com mais de 5 anos de experiência também vê o Requer um conhecimento prévio para criar e para entender e Não é possível demonstrar todos os detalhes como desvantagens principais. Já o grupo com 5 anos de experiência ou menos, vê o Requer um conhecimento prévio para criar e para entender(97,14%) e a dificuldade de criar(50,48%) como desvantagens principais.

3.6 Modelos Gráficos na Documentação de Software

O uso de modelos gráficos ajudam na compreensão de processos, design e arquitetura do software.

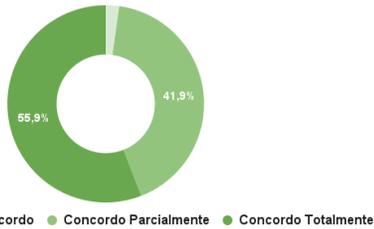


Figura 17: Os modelos gráficos ajudam na compreensão.

Modelos gráficos são imprescindíveis na documentação de um software.

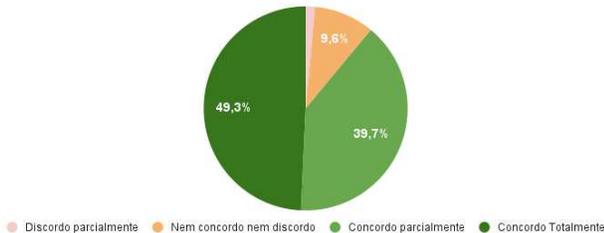


Figura 18: Modelos gráficos são imprescindíveis em uma documentação de software.

Nessa subseção abordaremos apenas a opinião de todos os respondentes. Em ambas perguntas: O uso de modelos gráficos ajudam na compreensão de processos, design e arquitetura do software; e os modelos gráficos são imprescindíveis na documentação de um software. Foi utilizada a Escala de Likert[10] para entender o nível de concordância com a afirmação dada.

Na figura 17 mostra que 97,8% dos respondentes concordaram que o uso de modelos realmente ajudam na compreensão de processos, design e arquitetura.

Na figura 18 mostra que 89% concordam que os modelos gráficos são imprescindíveis na documentação de um software. Esse é um número importante, mostra que tais modelos em conjunto com aspectos textuais ajudam a compreender melhor o software. Mas

diferentemente da figura 17, nesse caso houveram respostas que discordaram da afirmação(1,5%).

3.7 Modelos Gráficos Utilizados na Indústria

4.7.1 Utilizados

Modelos gráficos que você já utilizou/utiliza dentro do ambiente de trabalho do desenvolvimento de software

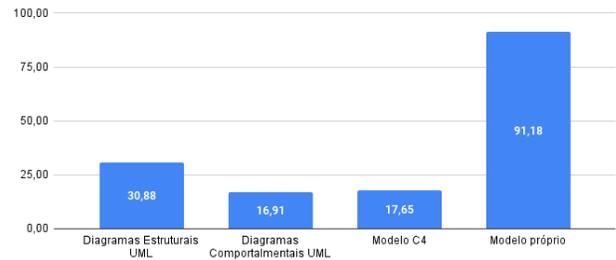


Figura 19: Modelos gráficos mais utilizados pelo público geral.

Modelos gráficos que você já utilizou/utiliza dentro do ambiente de trabalho do desenvolvimento de software.

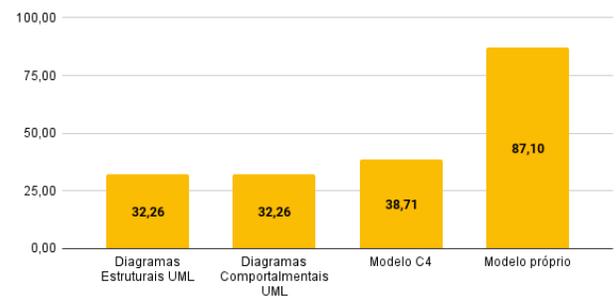


Figura 20: Modelos gráficos mais utilizados pelo público com mais de 5 anos de experiência.

Modelos gráficos que você já utilizou/utiliza dentro do ambiente de trabalho do desenvolvimento de software.

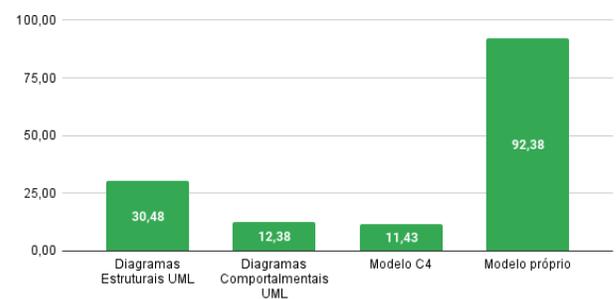


Figura 21: Modelos gráficos mais utilizados pelo público com 5 anos ou menos de experiência.

Para essa questão, os respondentes tinham a possibilidade de marcação de 5 opções de modelos gráficos, que seriam: Os principais tipos de modelo UML[11]; UML-Comportamentais[12], UML- Estruturais[13] e C4[14].

Também modelos gráficos próprios ou outros tipos de modelos gráficos.

O principal modelo gráfico citado foi o Modelo Próprio(91,18%). Já o segundo foi o Diagrama Estruturais(30,88%), em sequência o Modelo C4(17,65%) e em último o Diagrama Comportamental(16,91%).

Observando as figuras 20 e 21 nota-se que o grupo com mais de 5 anos de experiência tende a usar mais os Diagramas Comportamentais UML e o Modelo C4 em comparação com o grupo com menos de 5 anos de experiência.

Um ponto destacável é a utilização de modelos próprios pelos respondentes, inclusive entre aqueles que já possuem uma bagagem técnica e conhecem os mais variados tipos de modelos gráficos.

4.7.1 Modelos Relevantes e Imprescindíveis

Quais modelos gráficos você acha imprescindíveis dentro do desenvolvimento de software?

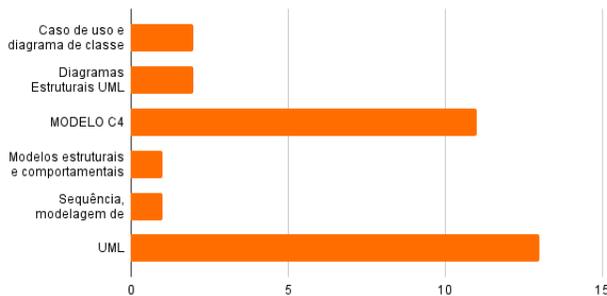


Figura 22: Modelos gráficos imprescindíveis para o público geral.

Quais modelos gráficos você acha relevantes dentro do desenvolvimento de software?

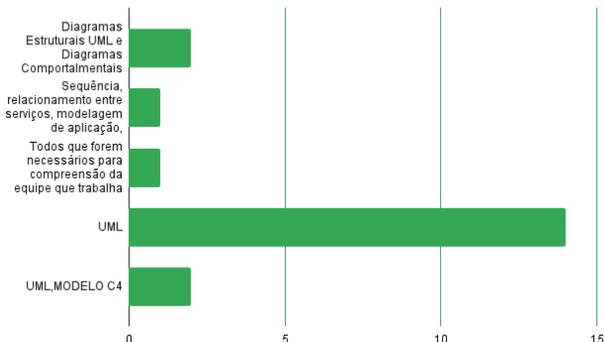


Figura 23: Modelos gráficos relevantes para o público geral.

Os Diagramas UML e o Modelo C4 são os modelos gráficos mais citados como imprescindíveis no contexto do desenvolvimento de software, vemos isso na figura 22. Vale notar que o UML citado nas respostas não é em referência a linguagem mas a um conjunto de um ou mais diagramas UML, o mesmo também afirmado para figura 23. Já na figura 23 a relevância dos Diagramas UML é vista de maneira clara em relação ao Modelo C4 e aos outros tipos

de modelos. Mas em ambas perguntas menos de 30% dos respondentes citaram algum modelo como relevante ou imprescindível.

4. CONSIDERAÇÕES FINAIS

Ao longo do artigo vemos que a utilização dos modelos tradicionais como UML e Modelo C4 não são mais tão adotadas como antes, apenas um terço dos respondentes disseram que utilizam esses modelos em seus trabalhos. O que é um número baixo comparado ao uso de modelos próprios que chega a 91% dos entrevistados, Também é importante notar que a criação de documentação não é o principal motivo de criação mas para demonstrar o fluxo de algo sem necessariamente virar documentação. O principal benefício desses modelos gráficos é que eles facilitam a compreensão e isso foi citado por todos os respondentes. Já a principal desvantagem é que para entender e criar os modelos é necessário um conhecimento prévio.

5. LIMITAÇÕES E TRABALHOS FUTUROS

Para consolidar as questões de pesquisa levantadas, é importante repetir as análises focando essencialmente em cada grupo em separado. Se necessário até diminuir a faixa de experiência dos respondentes para ter uma visão mais minuciosa de cada ano de experiência profissional dentro do desenvolvimento de software. Uma vez que o trabalho de pesquisa geral foi realizado as entrevistas seriam outro bom método de pesquisa para trabalhos futuros, focado em entender mais como os modelos tradicionais não estão atendidos as demandas atuais, entender a construção e se existe um padrão na construção desses modelos próprios, o que é levado em conta na sua criação, porque não utilizar os outros modelos prontos, se utilizam aspectos desses modelos já consolidados. Também seria importante identificar como as desvantagens dos modelos poderiam ser contornadas.

6. REFERÊNCIAS

- [1] Frederick Brooks. No Silver Bullet Essence and Accidents of Software Engineering. 1987 (4-7) <https://ieeexplore.ieee.org/document/1663532>
- [2] 2020. Cap. 4: Modelos - Engenharia de Software Moderna(Livro Digital). <https://engsoftmoderna.info/cap4.html> [Online; acessado em 08 de Agosto de 2021].
- [3] 2021. Formulário da Pesquisa. <https://forms.gle/hW2yM7X2Yg8wseAK8>. [Online; acessado em 05 de Setembro de 2021].
- [4] 2021. Dados Pesquisa 1. https://docs.google.com/spreadsheets/d/1A05-8wupndqj_1JzvpEQWHlatbO_AmyPZCL-Xbet1kA/edit?usp=sharing.

[Online; acessado em 05 de Setembro de 2021]. [Online; acessado em 05 de Setembro de 2021].

- [5] 2021. Análise dos dados 2. https://docs.google.com/spreadsheets/d/1AdvziLwnc8Uf2IIE0ArFhEA9_dsx_VClsqnsQqPNwC8/edit?usp=sharing. [Online; acessado em 05 de Setembro de 2021].
- [6] 2020. Eduardo Silva. Como saber se um desenvolvedor é Júnior, Pleno ou Sênior?. <https://rhtech.geekhunter.com.br/desenvolvedor-e-junior-pleno-ou-senior/>. [Online; acessado em 02 de Setembro de 2021].
- [7] 2020. Waldyr Felix. O que é um Tech Lead. <https://medium.com/checkpoint-di%C3%A1rio/o-que-%C3%A9-um-tech-lead-137fb85da02e>. [Online; acessado em 02 de Setembro de 2021].
- [8] 2020. O que faz um arquiteto de software?. <https://www.igti.com.br/blog/o-que-faz-um-arquiteto-de-software>. [Online; acessado em 25 de Julho de 2021].
- [9] 2021. Isabella Leal. O QUE É SER UM(A) QA?. <https://hackerculture.com.br/?p=1154>. [Online; acessado em 25 de Agosto de 2021].
- [10] 2019. Henrique Carvalho. O que é a Escala Likert e como aplicá-la. <https://vidadeproduto.com.br/escala-likert/> [Online; acessado em 15 de Agosto de 2021].
- [11] Alberto Manuel Rodrigues da Silva Carlos Alberto Escaleira Videira. UML, Metodologias e Ferramentas CASE. 2001. (111-118). http://www.cesarkallas.net/arquivos/livros/informatica/UML_Metodologias_e_Ferramentas_CASE_portugues_.pdf
- [12] Alberto Manuel Rodrigues da Silva Carlos Alberto Escaleira Videira. UML, Metodologias e Ferramentas CASE. 2001. (197-230). http://www.cesarkallas.net/arquivos/livros/informatica/UML_Metodologias_e_Ferramentas_CASE_portugues_.pdf
- [13] Alberto Manuel Rodrigues da Silva Carlos Alberto Escaleira Videira. UML, Metodologias e Ferramentas CASE. 2001. (165-192). http://www.cesarkallas.net/arquivos/livros/informatica/UML_Metodologias_e_Ferramentas_CASE_portugues_.pdf
- [14] 2021. The C4 model for visualising software architecture. <https://c4model.com/> [Online; acessado em 25 de Julho de 2021].