

GQM+PA: Um Framework baseado em Goal-Question-Metric e Pesquisa-Ação para Ensino de Medição de Software

Saori Costa¹, Allysson Alex Araújo¹, Emanuel Coutinho¹,
Francisco Gomes¹ e Jerffeson Souza²

¹Universidade Federal do Ceará (UFC)
Grupo de Estudos em Sistemas de Informação e Inovação Digital (GESID)

²Universidade Estadual do Ceará (UECE)

Abstract. *The practical understanding of software measurement proves to be relevant for students' awareness of quality assurance. However, aligning theory and practice in teaching such a discipline is challenging, thus requiring collaborative and engaging methodologies. This study proposes a framework, called GQM+PA, that merges Action Research method with the Goal-Question-Metric process for teaching software measurement. From a qualitative and quantitative perspective, multiple case studies were evaluated at the Federal University of Ceará (Campus Crateús). The results reveal that students were able to improve their knowledge and empirically perceive the value of software measurement, including software metrics and GQM.*

Resumo. *A compreensão prática da medição de software demonstra-se pertinente para conscientização dos alunos sobre a garantia de qualidade. Todavia, alinhar teoria e prática no ensino de tal disciplina reflete-se desafiador, requisitando, assim, metodologias colaborativas e engajadoras. Este estudo propõe um framework, denominado GQM+PA, o qual integra o método Pesquisa-Ação com o processo Goal-Question-Metric para ensino de medição de software. Sob uma perspectiva quali-quantitativa, avaliou-se múltiplos estudos de caso na Universidade Federal do Ceará (Campus Crateús). Resultados revelam que os alunos foram capazes de aprimorar seus conhecimentos e perceber empiricamente o valor da medição de software, incluindo métricas de software e GQM.*

1. Introdução

A medição de software tem uma intrínseca relação com a avaliação de métricas em prol da garantia da qualidade de um sistema [Sato 2007]. Assim, faz-se necessário o suporte adequado de métodos que apoiem a definição de quais métricas devem ser avaliadas, dentre os quais pode-se destacar o *Goal-Question-Metric* (GQM) [Basili 1994]. Entretanto, a compreensão e execução de processos de software como, por exemplo, a medição de software, está relacionada a diferentes características, incluindo a experiência prévia [Clarke and O'Connor 2012]. Nesse sentido, o ensino de qualidade de software desempenha um papel fundamental na formação acadêmica dos discentes e, conseqüentemente, demanda abordagens engajadoras para os estudantes [Villavicencio and Abran 2012]. Considerando tais premissas, percebe-se a proeminência em investigar práticas que contribuam para alinhar a teoria e prática sob o contexto do ensino de medição de software.

Diante da motivação previamente exposta, este estudo objetiva integrar as etapas que constituem o GQM às práticas da Pesquisa-Ação (PA), mesclando ambos em

um *framework* conceitual que oriente o ensino de medição de software. Segundo [Grundy and Kemmis 1982], a PA é um método de pesquisa amplamente difundido em trabalhos educacionais e envolve “estratégias de ação planejada que são implementadas e, a seguir, sistematicamente submetidas a observação, reflexão e mudança”. Justifica-se, portanto, a relevância em investigar a adoção de tal concepção, haja vista que viabiliza promover uma compreensão integrada, atraente e prática do processo de medição de software. Quanto ao escopo metodológico, o presente trabalho é pautado pelo tipo de pesquisa exploratória sob a forma de múltiplos estudos de casos [Yin 2017]. Em termos de análise, adota-se uma perspectiva quali-quantitativa em decorrência da necessidade em compreender elementos subjetivos adjacentes ao fenômeno sob investigação.

2. GQM+PA

Em essência, o GQM possui três níveis: i) Conceitual (*Objetivo*): é determinado para um objeto relativo a algum modelo de qualidade; ii) Operacional (*Questão*): perguntas para caracterizar a forma de avaliação e cumprimento do objetivo; iii) Quantitativo (*Métrica*): dados associados às questões com o propósito de encontrar uma forma quantitativa de respondê-las [Sato 2007]. Nesse sentido, inspirado no *framework* de PA definido por Kemmis *et al.* (2013), este trabalho propõe a integração do processo de GQM ao seu contexto, formando assim, uma visão única e integrada de ensino de medição de software, denominada **GQM+PA**. O GQM+PA é constituído de **4 etapas**: 1) Observação Preliminar; 2) Planejamento; 3) Implementação e Observação e 4) Reflexão. Acoplado a tais etapas, têm-se **8 atividades específicas**, sendo 4 relacionadas ao GQM e 4 oriundas da PA. Além disso, existem **3 papéis**: *Gerente*, *Time de Discentes* e *Monitor*. Conforme pode ser visto na Figura 1, inicia-se com uma observação preliminar para determinar o tipo de problema a ser resolvido. Na segunda etapa, realiza-se o planejamento sobre o GQM e as práticas de mensuração e aprendizagem. Na terceira etapa, concretizam-se as ações planejadas na etapa anterior, bem como a observação sobre o que foi executado. Finalmente, realiza-se uma avaliação e reflexão sobre o que ocorreu, incluindo o cumprimento dos objetivos. Portanto, ao integrar as atividades da PA (cinza) com o GQM (azul), proporciona-se que as etapas que constituem o GQM também se façam presentes no processo de ensino.

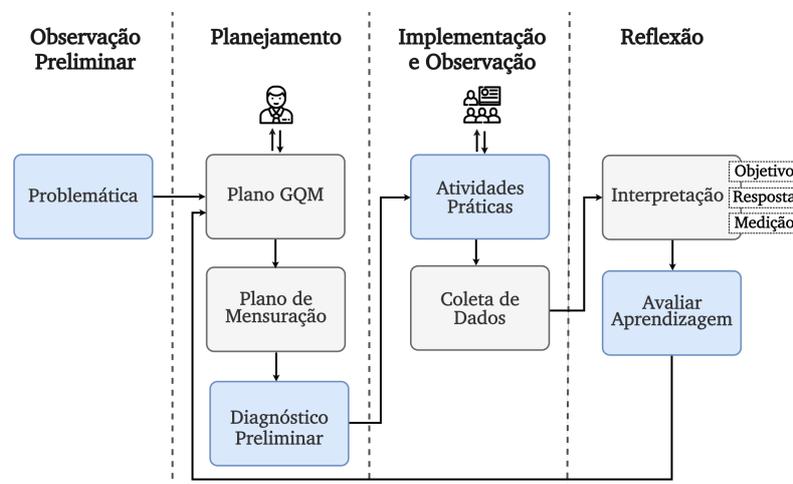


Figura 1. *Framework* GQM+PA - Visão Geral.

3. Estudo Empírico

A seguir será demonstrada a adoção do GQM+PA à luz do estudo empírico sob análise.

Observação Preliminar: a avaliação ocorreu no Núcleo de Práticas de Desenvolvimento de Sistemas¹ (NPDS) o qual encontra-se estabelecido na Universidade Federal do Ceará (*Campus* de Crateús). No NPDS não se emprega formalmente práticas de medição e adota-se a metodologia ágil *Scrum*, sendo estabelecido 10 dias para cada *sprint*. A alocação das atividades é definida pelo *Gerente* em conjunto com o *Time de Discentes*. Selecionou-se 3 projetos distintos durante 2 *sprints*, os quais viabilizam criar linhas de convergência e divergência sobre o material. Para cada projeto estavam alocados 2 estagiários (totalizando 6 alunos) os quais já haviam cursado a disciplina de Engenharia de Software. A autora deste artigo exerceu o papel de *Monitor*. Ela já havia concluído o estágio e dedicou-se exclusivamente ao acompanhamento metodológico.

Planejamento: uma vez feita uma avaliação prévia, torna-se necessário planejar a execução do GQM+PA através das seguintes atividades:

- **Plano GQM:** após entrevista semi-estruturada com o *Gerente*, identificou-se dois objetivos. O *Objetivo 1* trata de melhorar a precisão das estimativas de projeto e possui duas questões: a primeira “*Qual a precisão das estimativas do escopo do projeto?*”, avaliada pelas métricas Efetividade de Pontos, Velocidade de Pontos, Atividades Planejadas e Atividades Entregues. A segunda questão “*Qual a precisão das estimativas de esforço do projeto?*” foi avaliada pelas métricas Efetividade de Horas e Velocidade de Horas. Por sua vez, o *Objetivo 2* refere-se a aumentar a qualidade dos produtos liberados para uso e possui duas questões, cuja primeira foi: “*Qual a qualidade dos produtos antes de liberação para uso?*”, avaliada pelas métrica Complexidade Ciclométrica. A segunda “*Qual foi a aceitação das funcionalidades entregues?*” avaliada pelas métrica Atividades Aprovadas. Detalhes sobre a entrevista e métricas estão disponíveis na página de suporte².
- **Plano de Mensuração:** foram definidas rotinas diárias e de retrospectiva para os objetivos. Na Rotina Diária referente ao *Objetivo 1*, os alunos preencherão diariamente dados sobre o tempo executado e planejado das atividades. Ao final de cada *sprint*, os alunos realizarão a Rotina de Retrospectiva referente aos dados de estimativas. Por sua vez, na Rotina Diária referente ao *Objetivo 2*, os preencherão os dados sobre a qualidade do produto. No fim da *sprint*, realiza-se a Rotina de Retrospectiva para analisar a quantidade de *Atividades Aprovadas* pelo *Gerente*.
- **Diagnóstico Preliminar:** buscou-se fornecer uma base conceitual para que os alunos exerçam as rotinas por meio de um *workshop* de 2 horas. Adicionalmente, aplicou-se dois questionários semi-estruturados como instrumentos de avaliação. O primeiro objetivando metrificar o conhecimento prévio sobre medição de software e GQM, e o segundo quanto ao ambiente de desenvolvimento.

Implementação e Observação: após o processo de capacitação preliminar previamente contextualizado, as planilhas são disponibilizadas para que os alunos realizem as Rotinas Diárias e Rotinas de Retrospectiva. Através de tais rotinas e sob supervisão do *Monitor*, espera-se que, em um contexto prático de aprendizado, os alunos apresentem

¹<https://npds.crateus.ufc.br>

²http://gesid.github.io/papers/gqm_pa

uma melhor compreensão das métricas e o impacto na qualidade. Destaca-se também a responsabilidade do *Monitor* em acompanhar e interagir com os alunos a fim de lidar em conjunto com as problemáticas identificadas.

Reflexão: para analisar o processo de aprendizagem dos alunos, empregou-se, além da observação via *Monitor*, um terceiro questionário com o objetivo de verificar o nível de conhecimento após a adoção do GQM+PA, tornando possível comparar os resultados antes e após intervenção. Todas as planilhas referentes as rotinas de aprendizado, bem como os questionários encontram-se disponíveis na página de suporte.

4. Resultados e Análises

Dos resultados acerca do nível de conhecimento sobre **medição de software**, conforme a Figura 2(a), verifica-se que, antes de utilizar o GQM+PA, 5 (83%) alunos já haviam ouvido falar sobre medição de software, enquanto apenas 1 (16%) aluno alegou saber alguns conceitos. Ao *Monitor*, os alunos mencionaram que já tinham tido contato com medição na disciplina de Engenharia de Software, porém, nenhum aplicava formalmente. Após adoção do GQM+PA, nota-se que 66% dos alunos indicaram que agora sabem razoavelmente (3) ou sabem alguns conceitos (1). Um aluno destacou: *“Me prendi mais ao detalhe da complexidade e tentei não extrapolar os valores medianos, fazendo métodos menores, dividindo bem as funções”*. Apesar da maioria dos alunos terem evoluído nesse sentido, ainda registra-se 2 (33%) alunos que responderam que já ouviram falar, mesmo participando de um processo de medição onde diariamente coletava-se e analisava-se métricas. Isso reflete uma dificuldade em associar certas rotinas como medição de software.

A Figura 2(b) reflete as respostas sobre **métricas de software**. Nota-se que, antes de ser aplicado o GQM+PA, 4 (66%) alunos responderam que já ouviram falar sobre métricas, enquanto somente 1 (16%) relatou que não tinha nenhum conhecimento e 1 (16%) declarou que sabe alguns conceitos. Após a adoção do GQM+PA, constata-se que 82% dos alunos informaram que sabiam razoavelmente (4) ou sabiam alguns conceitos (1). Isso revela que a maioria dos alunos adquiriu mais conhecimentos sobre métricas através da prática diária. Apesar disso, verifica-se que um aluno respondeu que apenas ouviu falar sobre métricas, mesmo realizando diariamente a coleta. Quando questionados sobre adoção de métricas após uso do GQM+PA, os alunos, de maneira geral, reforçaram a importância das métricas no processo de desenvolvimento. Um aluno, por exemplo, ressaltou *“A possibilidade de acompanhar o rendimento de forma mais intrínseca no processo de desenvolvimento”*. Outro discente frisou que *“Houve uma mudança de atitude nas equipes que passaram a observar um pouco mais o andamento do seu trabalho”*.

Através da Figura 2(c) analisa-se o conhecimento sobre **GQM**. Antes de aplicar o GQM+PA, 5 (83%) alunos responderam que não tinham nenhum conhecimento sobre GQM e apenas 1 (16%) relatou que já ouviu falar. Após a aplicação do GQM+PA, verifica-se que 82% dos alunos passaram a definir que sabiam muito (1), sabiam razoavelmente (2) ou sabiam alguns conceitos (2). Nenhum aluno declarou não possuir nenhum conhecimento. Ao ser questionado sobre continuar adotando GQM+PA, um aluno reforçou que *“Sim, pois ele vai nos ajudar a nos auto-avaliarmos, facilitando o processo de análise dos próprios erros e proporcionando melhorias a partir disso.”*

Dentre os principais desafios identificados na observação do *Monitor*, destaca-se a incorporação da rotina diária de preenchimento das planilhas. Frequentemente os alunos

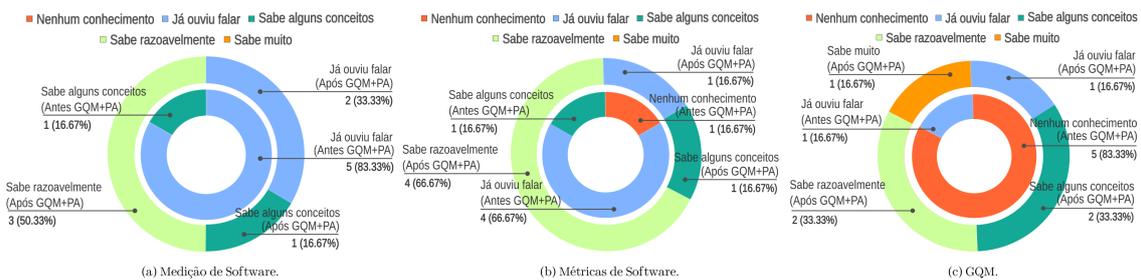


Figura 2. Nível de conhecimento sobre Medição, Métricas e GQM.

ainda esqueciam de preencher as planilhas, requisitando acompanhamento intenso.

5. Considerações Finais

Este trabalho propôs um *framework* (GQM+PA) o qual integra as etapas que constituem o GQM às práticas da Pesquisa-Ação. Dentre as contribuições, salienta-se: i) a disponibilização de um *framework* integrado para o aprendizado teórico e prático de medição de software; ii) viabilização de reflexão empírica aos discentes; e iii) todas as rotinas, questionários e planilhas desenvolvidas podem ser evoluídos como um ativo organizacional pelo NPDS e por outras instituições. Quanto à integração curricular, avalia-se como factível o uso em projetos atrelados à disciplinas específicas ou ao estágio.

Em relação às limitações, ressalta-se o tempo reduzido de observação, porém verificou-se que os resultados demonstram-se convergentes conforme aproximou-se do final do processo. Apesar da quantidade reduzida de alunos, destaca-se a qualificação do ambiente sob investigação o qual reflete particularidades reais. A análise de questões de múltipla escolha pode demonstrar-se restritivas quanto a percepção de aprendizado, porém, também adotou-se a observação e a análise de questões abertas com o intuito de capturar outros elementos subjetivos. Em termos de trabalhos futuros, almeja-se desenvolver um sistema de apoio que facilite a coleta, visualização e análise referente aos resultados obtidos, bem como investigar tal adoção em outros ambientes de ensino.

Referências

- Basili, V. R. (1994). Goal question metric paradigm. *Encyclopedia of software engineering*, pages 528–532.
- Clarke, P. and O'Connor, R. V. (2012). The situational factors that affect the software development process: Towards a comprehensive reference framework. *Information and Software Technology*, 54(5):433–447.
- Grundy, S. and Kemmis, S. (1982). Educational action research in Australia: The state of the art (an overview). *The action research reader*, 3:321–335.
- Kemmis, S., McTaggart, R., and Nixon, R. (2013). *The action research planner: Doing critical participatory action research*. Springer Science & Business Media.
- Sato, D. (2007). Uso eficaz de métricas em métodos ágeis de desenvolvimento de software. Master's thesis, Universidade de São Paulo, São Paulo.
- Villavicencio, M. and Abran, A. (2012). The necessary software measurement knowledge in software engineering education from the practitioners' point of view. In *25th IEEE Canadian Conference on Electrical and Computer Engineering*, pages 1–5. IEEE.
- Yin, R. K. (2017). *Case study research and applications: Design and methods*. Sage publications.