

# Explicação TCC - Atividade 2

---

22 de fevereiro de 2022

Arthur Rocha Amaral e Guilherme Oliveira Antônio

{ **arthur.amaral.1100245, gantonio** }@sga.pucminas.be

O seguinte projeto visa auxiliar os desenvolvedores a produzirem códigos com padrões de qualidade, definidos por repositórios, de forma mais rápida. Assim, garante segurança e padronização das entregas, e melhora a qualidade dos testes automatizados, afetando consecutivamente a manutenibilidade do software. De acordo com esse tema proposto para o Trabalho de Conclusão de Curso (TCC), a solução conta com a facilitação do processo de verificação de métricas de testes, a qual é desempenhada pela maioria dos desenvolvedores que aplicam uma metodologia de testes automatizados para garantir uma melhor qualidade de código. Desse modo, a ferramenta encurta o tempo entre análise de cobertura de testes e ajustes. Agilizando o processo de desenvolvimento de novas *features* e de correções. Sendo o público alvos desenvolvedores de diversos níveis da indústria de software.

O software se inclui no salão de ferramentas do Congresso Brasileiro de Software (CBsoft), o qual tem intuito de trazer benefícios aos desenvolvedores de software durante o processo de desenvolvimento e software. Trazendo benefícios que ajudam na abrangência dos testes e no controle de qualidade do projeto em desenvolvimento. Desse modo, é entendido que o software proposto se encaixa dentre as áreas da CBsoft na área de Teste de Software e Engenharia de Software. Tornando a aplicação proposta viável, visto que a mesma procura ajudar nos processos de desenvolvimento, gerenciamento e manutenção de projetos. Além disso, ajuda na qualidade da construção de testes medindo a cobertura do projeto e exibindo em tempo real para o desenvolvedor no próprio ambiente de desenvolvimento.

O sistema proposto é uma extensão para o Visual Studio Code (VS Code), o qual tem o intuito de proporcionar aos desenvolvedores uma melhor experiência no desenvolvimento com testes automatizados. Dessa forma, exibe os dados de cobertura de teste, possibilita a configuração por repositório de níveis de aceitação de testes de automatizados e pode ser incluídos em processos de publicação e versionamento, garantindo a qualidade dos entregáveis em tempo desenvolvimento. Para isso o programa conta com a leitura do arquivo de *line coverage* (lcov) gerado por *frameworks* de testes e será coletado em tempo real para exibição do *status* atual do desenvolvimento. Também é possível ler o arquivo de configuração na raiz do projeto para definir os critérios de aceitação do projeto e as variáveis necessárias para a configuração da extensão. Outros pontos importantes de facilitação do desenvolvimento incluem: o destaque das linhas de código cobertas ou não em um arquivo no editor; obter resumos de dados coberturas na barra lateral do Ambiente de desenvolvimento integrado *Integrated Development Environment* (IDE ) incluindo porcentagem de cobertura total do código; porcentagem de cobertura de código das linhas criadas; listagem da linha sem cobertura; porcentagem de condicionais cobertas por testes; listagem das condicionais sem cobertura; redirecionar automaticamente para uma linha de código sem cobertura dentre as linhas indicadas na barra lateral da IDE; definir a *branch* de referência para análise do código. Para mais, a exibição dos dados para o usuário é utilizado a integração com a *Application Programming Interface*(API) VSCode chamado *Source*

*Control API*. Além da integração da Api do VSCode é utilizado a Api do Git, a qual permiti o bloqueio ou aceitação do conteúdo desenvolvido seguindo os parâmetros de qualidade definidos anteriormente.

O sistema proposto se assemelha com outra extensão do VSCode que se chama Code Coverage. Essa aplicação também foi desenvolvida para auxiliar os desenvolvedores, escrita em TypeScript (TS), possui duas das funcionalidades que se assemelham com as propostas por essa aplicação, sendo elas: destaque das linhas cobertas no arquivo no editor; exibição da porcentagem de cobertura por arquivo. Além disso, existe outra extensão chamada Coverage Gutters, a qual tem o objetivo único de destacar as linhas cobertas por testes no arquivo no editor, que não é o nosso objetivo principal. Essas duas extensões utilizam arquivos de lcov para exibir os dados de cobertura e assim, conseguem atender a qualquer linguagem que utiliza por esta categoria de arquivo para gerar os dados de cobertura. Já a aplicação proposta, também utiliza tal padrão de relatório de cobertura para gerar os dados, mas tem somente como linguagens suportadas oficialmente o JavaScript (JS) e o TS.

Além das extensões já citadas, outras aplicações fazem esse tipo de análise. Como o SonarQube, o qual utiliza também de outros fatores para definir a qualidade do código, destacando os dados através da diferença entre *branches* do Git, além da cobertura de código com base no arquivo de lcov. Possui integração com a interface de desenvolvimento em questã o VSCode. Porém, apenas na parte de segurança de código SonarQube Project Status, a qual exibe métricas de *bugs*, *code smells* e vulnerabilidades encontradas pela plataforma. Dessa forma, a principalmente relação são as métricas de cobertura, as quais são exibidas de forma mais, devido à necessidade de integração com um servidor que esteja executando a aplicação de análise de código. A aplicação proposta neste trabalho irá analisar no *client side*, o arquivo com os dados de resultado dos testes sem a necessidade de execução de um instância da aplicação para análise do código.

Devido aos fatores apresentados, a aplicação proposta neste documento tende a trazer os benefícios que visam agilizar o processo de desenvolvimento. Facilitando a padronização da qualidade do software ainda no ambiente de desenvolvimento. Apesar de a cobertura de código ser uma métrica básica a ser levada em consideração, um processo de padronização dessa métrica garante o desenvolvimento de aplicações estáveis com menos chance de alteração de requisitos indesejadamente. Além disso, pelos fatores levantados é fato que a utilização dessa extensão proposta substitui outras aplicações que executam em servidores para garantir essas métricas, porém trazendo para dentro da IDE o caminho para se obter esses dados é muito menor.