

## ENUNCIADO DO PROJETO DE ENGENHARIA DE SOFTWARE I

TÍTULO	Aplicação para avaliação da qualidade de deteção de defeitos de desenho em projetos de software
DESCRIÇÃO	<p>Para o desenvolvimento deste trabalho será dado um conjunto de métricas sobre código fonte Java extraídas de um projeto de referência (número de linhas de código [LOC], complexidade ciclomática do método [CYCLO], acessos do método a métodos de outras classes [ATFD], acessos do método a atributos da própria classe [LAA]). Estas métricas são usadas para efeitos de deteção de defeitos de desenho no código fonte do software.</p> <p>A deteção de defeitos é realizada através da aplicação de regras que envolvem cálculo sobre estas métricas. No caso concreto do defeito designado por <i>long_method</i>, a regra a aplicar na aplicação a desenvolver pelo grupo é:</p> <p><i>SE</i> (LOC &gt; 80 E CYCLO &gt; 10) <i>ENTÃO</i> <i>is_long_method</i> = TRUE <i>SENÃO</i> <i>is_long_method</i> = FALSE</p> <p>Os valores 10 e 80 no cálculo acima, correspondem aos limites (<i>thresholds</i>) a partir dos quais o defeito está presente.</p> <p>Note que estes valores (<i>thresholds</i>) podem ser configurados como parâmetros nas ferramentas de deteção de defeitos no software, criando assim a possibilidade de as equipas de desenvolvimento de software definirem as suas próprias regras de deteção de defeitos.</p> <p>Será disponibilizada informação sobre a existência (ou não) do defeito de desenho de software designado de <i>long_method</i>, para cada método presente no projeto Java de referência, bem como a classificação que as ferramentas de deteção de defeitos no software iPlasma e PMD fazem para cada um dos métodos do projeto.</p> <p>Note que as ferramentas de deteção de defeitos em código fonte adotam diferentes estratégias no processo de deteção de defeitos no software (incluindo estratégias que não são baseadas em regras) e por isso cometem erros de classificação, ou seja, por vezes identificam defeitos no código fonte, que na realidade não existem, outras vezes não detetam defeitos onde efetivamente os defeitos estão presentes.</p> <p>As métricas extraídas sobre o código fonte do projeto Java de referência, a classificação de defeitos realizada por cada ferramenta (iPlasma e PMD) e os defeitos de software do tipo <i>is_long_method</i> efetivamente presentes no projeto, são disponibilizados num ficheiro Excel com nome <i>Long-Method.xlsx</i>.</p> <p>O ficheiro Excel disponibilizado contém 12 colunas, correspondentes a:</p> <ul style="list-style-type: none"><li>- <i>MethodID</i>, correspondente à identificação do método do projeto Java;</li><li>- <i>package</i>, correspondente à identificação do package onde se encontra o método;</li><li>- <i>class</i>, correspondente à identificação da class onde se encontra o método;</li><li>- <i>method</i>, correspondente ao nome do método;</li><li>- <i>LOC</i>, correspondente ao número de linhas de código do método;</li><li>- <i>CYCLO</i>, correspondente à complexidade ciclomática do método;</li><li>- <i>ATFD</i>, correspondente aos acessos que o método faz a métodos de outras classes;</li><li>- <i>LAA</i>, correspondente aos acessos do método a atributos da própria classe;</li><li>- <i>is_long_method</i>, correspondente à classificação correta do defeito <i>is_long_method</i> no método, ou seja, se o defeito <i>long_method</i> se encontra presente ou não no método;</li><li>- <i>iPlasma</i>, correspondente à classificação que a ferramenta iPlasma faz sobre a presença (ou não) do defeito <i>long_method</i> no método;</li><li>- <i>PMD</i>, correspondente à classificação que a ferramenta PMD faz sobre a presença (ou não) do defeito <i>long_method</i> no método;</li><li>- <i>is_feature_envy</i>, correspondente à classificação correta do defeito <i>feature_envy</i> no método, ou seja, se o defeito</li></ul>

	<p><i>feature_envy</i> se encontra presente ou não no método.</p> <p>O grupo de trabalho de ES1 deverá desenvolver uma aplicação em Java que permita ao utilizador avaliar a qualidade (número de acertos e de erros na classificação de defeitos) das ferramentas iPlasma e PMD, na deteção de defeitos de desenho de software, a partir dos dados fornecidos para o trabalho neste enunciado e no ficheiro <i>Long-Method.xlsx</i>.</p> <p>Para além da avaliação da qualidade das ferramentas de deteção de defeitos no desenho de software iPlasma e PMD, a aplicação deverá também ser capaz de efetuar a deteção de (novos) defeitos, que podem ser definidos pelo utilizador através da interface gráfica da aplicação (GUI).</p> <p>Deverá poder ser visualizada na GUI, de forma comparativa (textual/tabular/gráfica), a qualidade das ferramentas iPlasma e PMD. Deverá também ser possível através da GUI, criar/editar/visualizar as regras/<i>thresholds</i> criados/definidos pelo utilizador para a deteção de defeitos no software, bem como visualizar a qualidade das regras/<i>thresholds</i> definidos pelo utilizador.</p> <p>A qualidade de deteção de defeitos das ferramentas, é medida neste contexto considerando o número de acertos e o número de erros verificados na deteção de defeitos, relativamente ao projeto Java disponibilizados como referência para o trabalho, descrito no ficheiro Excel.</p> <p>A aplicação deverá por isso mostrar através da GUI o número de <i>Defeitos Corretamente Identificados</i> (DCI), o número de <i>Defeitos Incorretamente Identificados</i> (DII), o número de <i>Ausências de Defeitos Corretamente Identificadas</i> (ADCI) e o número de <i>Ausências de Defeitos Incorretamente Identificadas</i> (ADII). Estes indicadores são calculados da seguinte forma:</p> <ul style="list-style-type: none"> <li>- DCI, quando o valor da coluna correspondente à ferramenta (PMI ou iPlasma) é TRUE e a coluna <i>is_long_method</i> também é TRUE;</li> <li>- DII, quando o valor da coluna correspondente à ferramenta (PMI ou iPlasma) é TRUE e a coluna <i>is_long_method</i> é FALSE;</li> <li>- ADCI, quando o valor da coluna correspondente à ferramenta (PMI ou iPlasma) é FALSE e a coluna <i>is_long_method</i> também é FALSE;</li> <li>- ADII, quando o valor da coluna correspondente à ferramenta (PMI ou iPlasma) é FALSE e a coluna <i>is_long_method</i> é TRUE.</li> </ul> <p>Pretende-se que sejam implementadas as seguintes funcionalidades na aplicação a desenvolver pelo grupo:</p> <ol style="list-style-type: none"> <li>1) Importação e visualização do ficheiro Excel através da GUI da aplicação;</li> <li>2) Incluir na GUI a possibilidade do utilizador definir os valores limites (<i>thresholds</i>) para as métricas envolvidas na deteção dos defeitos <i>long method</i> e <i>feature envy</i>;</li> <li>3) Permitir através da GUI que o utilizador possa definir regras e <i>thresholds</i> para deteção de defeitos, isto é, o utilizador escolhe as métricas a serem usadas na regra (entre as métricas já identificadas no ficheiro Excel disponibilizado), os <i>thresholds</i> e os operadores lógicos (AND ou OR) que pretende usar para formar expressões lógicas com as métricas.</li> </ol> <p>Exemplo de uma regra para deteção do defeito <i>feature envy</i>:</p> <p><i>SE ( ATFD &gt; 4 E LAA &lt; 0.42 ) ENTÃO is_feature_envy = TRUE SENÃO is_feature_envy = FALSE</i></p> <ol style="list-style-type: none"> <li>4) A deteção de defeitos realizada pela aplicação é baseada nas regras e <i>thresholds</i> definidos nos pontos 2 e 3, utilizando as métricas importadas do ficheiro Excel;</li> <li>5) Quando se efetuar a deteção dos defeitos, os resultados devem ser apresentados na GUI, com a indicação do MethodID, a ferramenta usada para a deteção e se foram ou não detetados defeitos;</li> <li>6) Devem ser mostrados na GUI os indicadores de qualidade verificados na deteção dos defeitos descritos acima (DCI, DII, ADCI, ADII) para as ferramentas iPlasma, PMD e para as regras/<i>thresholds</i> criadas/definidas pelo utilizador.</li> </ol>
<b>ATIVIDADES GENÉRICAS ENVOLVIDAS NA REALIZAÇÃO DO PROJETO</b>	<ul style="list-style-type: none"> <li>• Desenvolvimento do projeto de software seguindo o método ágil Scrum;</li> <li>• Desenvolvimento de software em Java usando o IDE Eclipse;</li> <li>• Criação e utilização de repositório de software git na plataforma github.com para suporte ao trabalho</li> </ul>

	<p>colaborativo do grupo e para a entrega do projeto;</p> <ul style="list-style-type: none"> <li>• Acompanhamento e gestão do projeto com recurso à plataforma Trello;</li> <li>• Acompanhamento e gestão da qualidade do projeto de software seguindo as boas práticas de qualidade para os projetos de software (indicadas abaixo no enunciado);</li> </ul>
<b>TECNOLOGIAS ENVOLVIDAS NA REALIZAÇÃO DO PROJETO</b>	<p>Ambiente de desenvolvimento: Eclipse IDE;</p> <p>Linguagens de programação: JAVA;</p> <p>APIs de integração com as várias fontes de informação/serviços: A serem escolhidas pelos grupos de trabalho;</p> <p>Plugins Eclipse: EclEmma, Modelgoon, Software Development Process Mining.</p>
<b>RESULTADOS ESPERADOS (DELIVERABLES)</b>	<p>Os elementos a disponibilizar na entrega do projeto (<i>deliverables</i>) são:</p> <ul style="list-style-type: none"> <li>• Todo o software desenvolvido deve ser disponibilizado na plataforma github.com através do repositório do grupo de trabalho com um identificador que siga o formato ES1-2019-Turma-NumeroDoGrupo (por exemplo, o grupo 99 da turma EIC1 deverá criar um repositório com a identificação ES1-2019-EIC1-99 na plataforma github);</li> <li>• A pasta raiz do repositório deverá conter um ficheiro de texto "README.md" com a identificação do grupo, identificação dos membros do grupo (nome e número), identificação e descrição de erros e funcionalidades não implementadas ou incompletas;</li> <li>• Documentação Javadoc do software desenvolvido pelo grupo de trabalho;</li> <li>• Testes unitários (JUnit) relativos ao software desenvolvido;</li> <li>• Relatório de cobertura de testes relativo à parte do software desenvolvido para o projeto, gerado com o plugin Eclipse – EclEmma, indicando sucesso para todos os testes realizados e uma cobertura de testes superior a 75%, em pelo menos duas métricas de cobertura de testes;</li> <li>• Para efeitos de entrega do projeto, deve ser associada a <i>tag DefectsDetection-1.0</i> ao último <i>commit</i> feito no <i>branch master</i> do repositório github do grupo;</li> <li>• Ficheiros CSV e Json resultantes do plugin Software Development Process Mining, e presentes na diretoria do seu Workspace: "Workspace"\.metadata\.plugins\iscte.analytics.plugin.eclipse;</li> <li>• Para efeitos de entrega dos elementos relativos à gestão do projeto da UC, será considerada a organização e atividades registadas pelo grupo de trabalho ao longo do projeto/semestre nas plataformas GitHub e Trello.</li> </ul>
<b>MÉTODO DE AVALIAÇÃO</b>	<p>O projeto prático será avaliado de acordo com os seguintes parâmetros de avaliação:</p> <ul style="list-style-type: none"> <li>• 20%, foi seguido corretamente o método ágil (Scrum) adotado para o projeto;</li> <li>• 20%, foram implementadas todas as funcionalidades no software solicitado;</li> <li>• 20%, foram realizados os testes unitários com cobertura mínima de 75%, em pelo menos duas métricas de cobertura de testes;</li> <li>• 20%, foram seguidas as boas práticas de desenho e implementação de software (modularidade, <i>code conventions</i>, JavaDocs, etc.) e as indicações de entrega do projeto;</li> <li>• 20% - Qualidade técnica do trabalho.</li> </ul> <p>A avaliação do trabalho será realizada sobre o projeto como um todo, havendo, no entanto, avaliações individuais para cada membro do grupo de trabalho.</p> <p>As avaliações individuais serão realizadas de acordo com as apresentações (de progresso e final) do projeto, as atividades registadas nas ferramentas de gestão do projeto e de acordo com evidências das atividades planeadas, tarefas atribuídas aos membros do grupo, execução e progressos do projeto registados.</p> <p>Os projetos serão objeto de verificação e comparação para efeitos de deteção de cópia de trabalhos. As ferramentas de gestão de projeto permitirão a verificação cronológica e auditoria do progresso de cada projeto.</p> <p>A deteção de práticas fraudulentas previstas nos regulamentos do ISCTE-IUL levará ao desencadear das correspondentes sanções aplicáveis.</p>