PUC Minas

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS

Atividade Prática

Teste de Software

Detecção de Bad Smells e Refatoração Segura

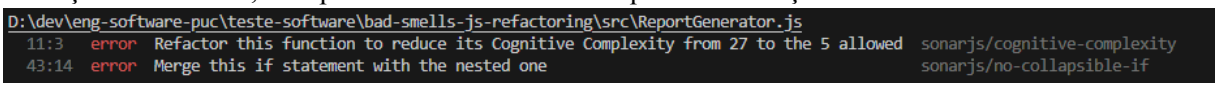
João Marcos de Aquino Gonçalves

1336608

Belo Horizonte 2025

Bad Smells detectados:
Durante a execução do sonarls + eslist, foi notado dois smells, sendo eles:
Cognitive Complexity, ou seja, Complexidade Cognitiva, que ocorre quando se tem uma estrutura de comparação confusa, com ifs, elses e fors. Seu objetivo é medir se o código está complicado além do necessário para cumprir seu propósito.
Collapsable If é um tipo de smell apontando que há muitos ifs aninhados que poderiam ser combinados em uma única condição
Basicamente, a função estava repetindo linhas e complicando além do necessário para garantir a elaboração de 4 tipos de reports: O report de usuário comum em CSV e HTML e o report de usuário admin em CSV e HTML.

Relatório da ferramenta:
A ferramenta foi extremamente útil para detectar onde haviam os bad smells no código, em sua execução no terminal, ela apontou os smells e linhas que eles começavam.



Além disso, ela também sugeriu mudanças diretamente em linhas detectadas no arquivo, provendo uma resolução mais direta e eficiente.

Processo de Refatoração:
Para refatorar, utilizei os metodo Extract Method para separar em 4 sub funções, cada uma para gerar um dos tipos de relatórios que a função se propõe a gerar, e para suas chamadas, utilizei o método Extract Variable para simplificar a deteção de tipo de arquivo e tipo de usuário.
A seguir, um exemplo de refatoração do método para Gerar report de usuário Admin em CSV.
Antes:

```
generateReport(reportType, user, items) {
    let report = '';
    let total = 0;

    // --- Seção do Cabeçalho ---
    if (reportType === 'CSV') {
      report += 'ID,NOME,VALOR,USUARIO\n';
    } else if (reportType === 'HTML') {
      report += '<html><body>\n';
      report += '<h1>Relatório</h1>\n';
      report += `<h2>Usuário: ${user.name}</h2>\n`;
      report += '<table>\n';
      report += '<tr><th>ID</th><th>Nome</th><th>Valor</th></tr>\n';
    }
```

```
    // --- Seção do Corpo (Alta Complexidade) ---
    for (const item of items) {
      if (user.role === 'ADMIN') {
        // Admins veem todos os itens
        if (item.value > 1000) {
          // Lógica bônus para admins
          item.priority = true;
        }

        if (reportType === 'CSV') {
          report +=
`${item.id},${item.name},${item.value},${user.name}\n`;
          total += item.value;
        } else if (reportType === 'HTML') {
          const style = item.priority ? 'style="font-weight:bold;"' :
'';
          report += `<tr
${style}><td>${item.id}</td><td>${item.name}</td><td>${item.value}</td>
</tr>\n`;
          total += item.value;
        }
      } else if (user.role === 'USER') {
        // Users comuns só veem itens de valor baixo
        if (item.value <= 500) {
          if (reportType === 'CSV') {
            report +=
`${item.id},${item.name},${item.value},${user.name}\n`;
            total += item.value;
          } else if (reportType === 'HTML') {
            report +=
`<tr><td>${item.id}</td><td>${item.name}</td><td>${item.value}</td></tr
>\n`;
            total += item.value;
          }
        }
      }
    }

    // --- Seção do Rodapé ---
    if (reportType === 'CSV') {
      report += '\nTotal,,\n';
      report += `${total},,\n`;
```

```
    } else if (reportType === 'HTML') {
      report += '</table>\n';
      report += `<h3>Total: ${total}</h3>\n`;
      report += '</body></html>\n';
    }

    return report.trim();
  }
}
```

Depois:

```
 generateAdminReportCSV(report, total, user, items) {
        // --- Seção do Cabeçalho ---
        report += 'ID,NOME,VALOR,USUARIO\n';
        // --- Seção do Corpo (Alta Complexidade) ---
        for (const item of items) {
            // Admins veem todos os itens
            if (item.value > 1000) {
                // Lógica bônus para admins
                item.priority = true;
            }
            report +=
`${item.id},${item.name},${item.value},${user.name}\n`;
            total += item.value;
        }
        // --- Seção do Rodapé ---
        report += '\nTotal,,\n';
        report += `${total},,\n`;
        return report.trim();
    }
generateReport(reportType, user, items) {
        let report = '';
        let total = 0;
        const isCSV = reportType === 'CSV';
        const isAdmin = user.role === 'ADMIN';

        if (isAdmin) {
            if (isCSV) {
                this.generateAdminReportCSV(report, total, user,
items);
            }
            this.generateAdminReportHTML(report, total, user, items);

        }
```

```
        // User comum
        if (isCSV) {
            this.generateUserReportCSV(report, total, user, items);
        }
        this.generateUserReportHTML(report, total, user, items);

    }
```

Conclusão:
Graças a essa ferramenta, foi possível medir de forma rápida e eficiente pontos na função que apresentavam complexidade exagerada e qualidade ruim de código, é notável como apesar da refatoração ter aumentado a quantidade de linhas da função, simplificou sua leitura e seu entendimento.