**Relatório de Verificação**

Análise Estática de Código

Nome:Elbert Jean dos Santos\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Turma:5 Semestre\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Data: 01/09/2025­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Projeto N - <Openai Agents Python> - <https://github.com/openai/openai-agents-python>**

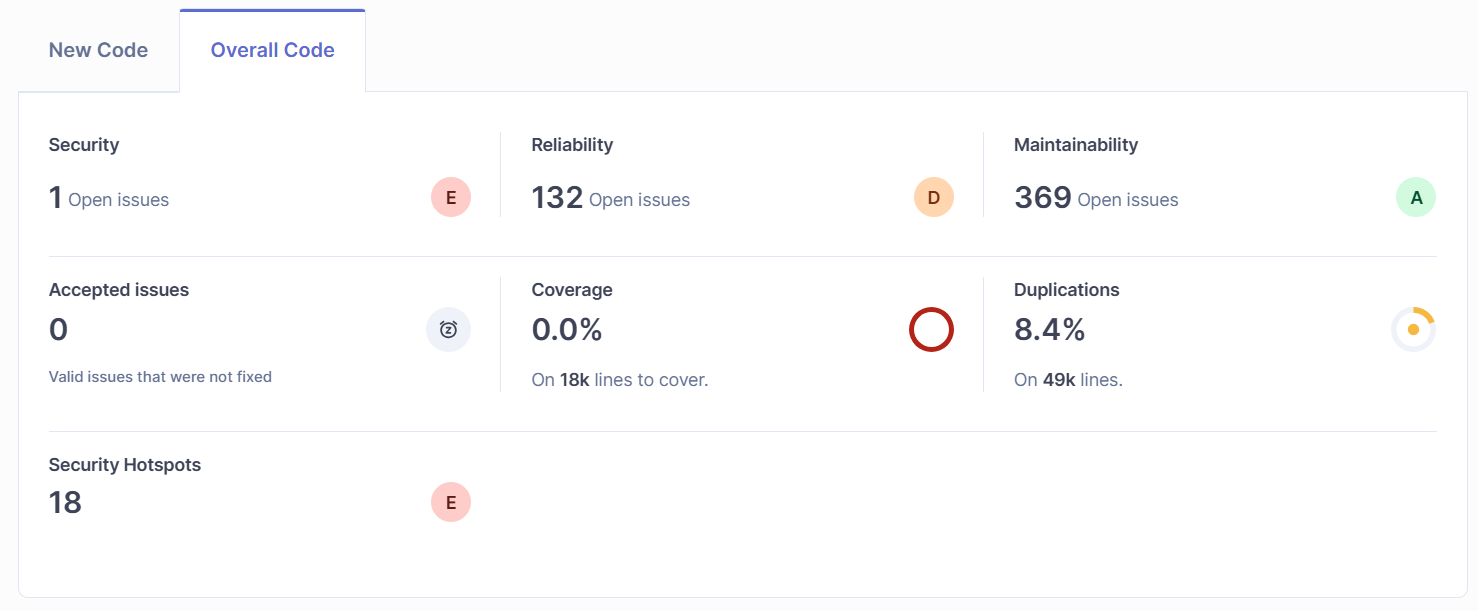
1. **Detalhamento dos Passos de Execução da Análise**
   1. Incluir os detalhes dos passos executados para processar a análise estática do projeto (não precisa incluiros passos para subir o SonarQube Server)

**R:** Após clonar o projeto do GitHub, é necessário criar um arquivo chamado **sonar-project.properties** na raiz do projeto. Esse arquivo contém as configurações necessárias para que o Sonar identifique o projeto e realize o upload do projeto para análise.

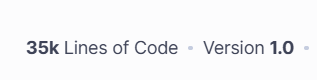
1. **Visão Sumarizada (Overall)**
   1. Análise inicial (v0) contemplando um quadro com as principais métricas/scores apresentados: Security, Reliability, Maintainability, Accepted Issues, Coverage, Duplications e Security Hotspots.

|  |  |
| --- | --- |
| **Métrica** | **Valor** |
| Security | 1 |
| Reliability | 132 |
| Maintanability | 369 |
| Accepted Issues | 0 |
| Coverage | 0.0% |
| Duplications | 8.4% |
| Security Hostpots | 18 |

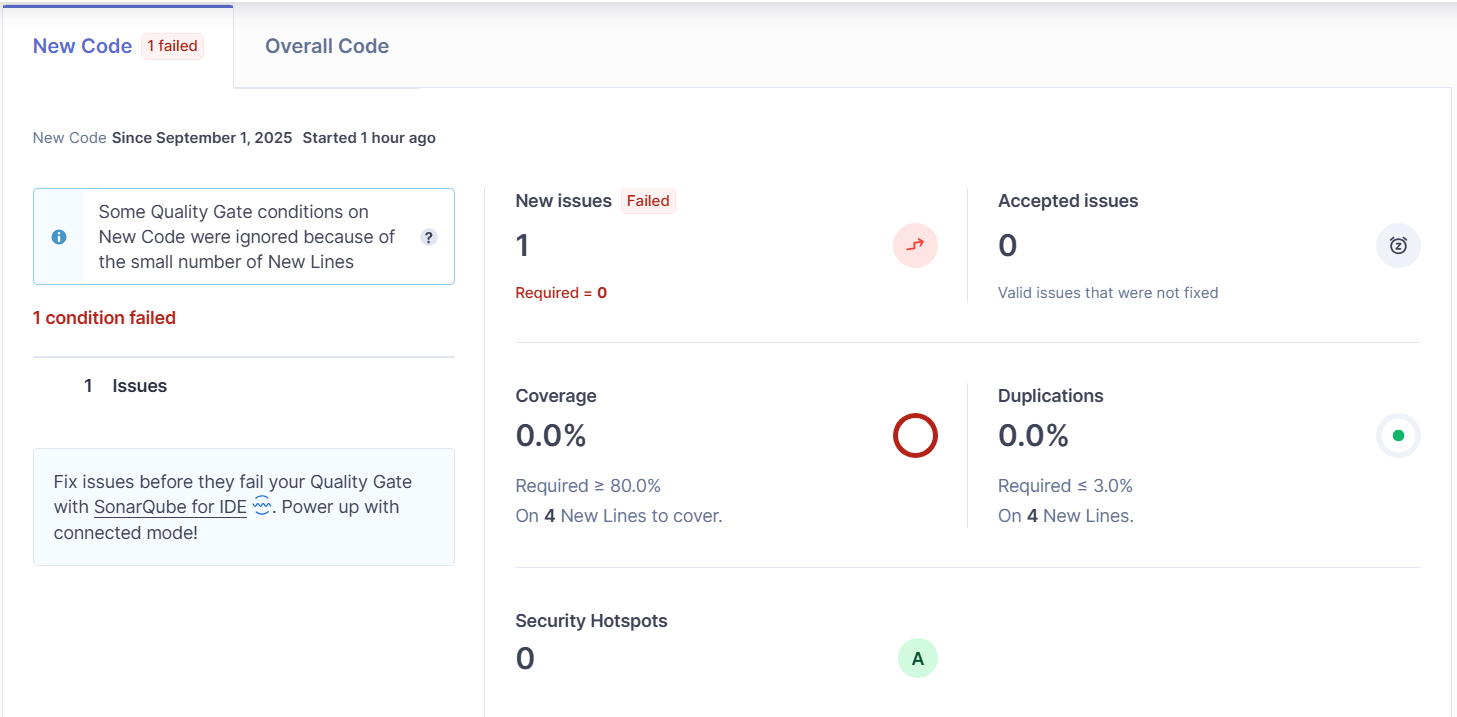
* 1. Incluir uma evidência da visão sumarizada da aba Overwall



* 1. Incluir a quantidade total de linhas de código no projeto



1. **Visão Sumarizada (New Code Lines)**
   1. Realizar de 2 a 3 alterações pontuais no projeto (fix de issues inferidas na primeira análise) e na sequencia processar por uma segunda vez a análise estática
   2. Incluir uma evidência da visão sumarizada da aba New Code



1. Criar 2 Quadros Quantitativos Detalhados – Overall VS New Code Lines
   1. Itens do quadro: Security, Reliability, Maintainability, Accepted Issues e Security Hotspots

|  |  |  |
| --- | --- | --- |
| **Métrica** | **Overall** | **New Code** |
| Security | 1 | 1 |
| Reliability | 132 | 132 |
| Maintainability | 369 | 368 |
| Accepted Issues | 0 | 0 |
| Security Hotspot | 18 | 18 |

* 1. Separar a quantidade identificada por grau de severidade: Blocker, High, Medium, Low e Info.

|  |  |
| --- | --- |
| **Grau de severidade** | **Quantidade** |
| Blocker | 1 |
| High | 122 |
| Medium | 259 |
| Low | 110 |
| Info | 9 |

* 1. Separar a quantidade identificada por Clean Code Attribute: Consistency, Intentionality, Adaptability e Responsability

|  |  |
| --- | --- |
| **Clean Code Attribute** | **Quantidade** |
| Consistency | 102 |
| Intentionality | 254 |
| Adaptability | 51 |
| Responsability | 1 |

1. Estratégia de Redução de *Issues* Identificadas

Estabelecer uma estratégia gradativa que não deve ocupar mais de 20% de esforço de um Time de 5 desenvolvedores full-time (5 x 8) para um TimeBox de uma Sprint de 3 semanas, visando atender o Quality Gate abaixo:

* Nr. Maintainability (Blocker e High) igual a 0
* Nr. Securities (Blocker) igual a 0
* Nr. Security Hotspots igual a 0
* % de Duplicidade de Linhas de Código igual a 0
* % de Cobertura de Código >= 80%

Detalhar a estratégia e projeção da quantidade de semanas necessárias para atingir o Quality Gate.

**Resposta:**

Levando em consideração os parâmetros abaixo:

|  |  |  |  |
| --- | --- | --- | --- |
| **Quality Gate** | **Quantidade** | **Tempo para cada task** | **Tempo total** |
| (Blocker e High Nr. Maintainability (Blocker e High) igual a 0 | 97 | 3h | 291 horas |
| Nr. Securities (Blocker) igual a 0 | 1 | 1h | 1 hora |
| Nr. Security Hotspots igual a 0 | 1 | 1h | 1 hora |
| % de Duplicidade de Linhas de Código igual a 0 | 38 | 4h | 152 horas |
| % de Cobertura de Código >= 80%) | 13.302 linhas na pasta src | 250 linhas de código por desenvolvedor (1250 linhas por semana) | 424 horas |

Considerando que o time de desenvolvimento tenha 200h de trabalho semanal, assumindo a estratégia de utilizar 20% desse tempo, temos 40 horas semanais (8 horas por desenvolvedor) para atingir o quality gate.

Somando todas as horas das atividades, temos 869 horas e considerando que cada Sprint tenha 3 semanas, teremos 7 sprints (120h para cada sprint).

Portanto, uma boa separação das atividades seria:

|  |  |
| --- | --- |
| **Sprint** | **Descrição:** |
| Sprint 1 | 99.17% para Block e High (119 horas)  1h para Securities Blocker (1 hora) |
| Sprint 2 | 100% para Block e High (120 horas) |
| Sprint 3 | 43.33% para Block e High (52 horas) 56.67% para % de duplicidade (68 horas) |
| Sprint 4 | 70% para % de duplicidade (84 horas) 30% para cobertura de teste (36) |
| Sprint 5 | 100% para cobertura de teste (120 horas) |
| Sprint 6 | 100% para cobertura de teste (120 horas) |
| Sprint 7 | 100% para cobertura de teste (120 horas) |
| Sprint 8 | 28 horas |