Metrici software și ingineria calității



**Overview Pyramid**

**Student:** Draghici Andreea-Maria

**Specializare:** Inginerie Software

**Grupa:** IS1.B

**Overview:**

**I used the next metrics:**

1. LOC ( lines of code )

2. CLOC ( lines of comment )

3. NCLOC ( lines of non-comment )

4. NOM ( number of methods )

5. C ( number of classes in each package )

6. NOI ( number of interfaces )

7. NOC ( number of direct subclasses of each class that occur in the project )

8. NOSC ( number of static classes )

9. WMC ( weighted method complexity )

10. BUGS ( average bugs per class )

11. VIOLATIONS ( problems per class / errors or warnings )

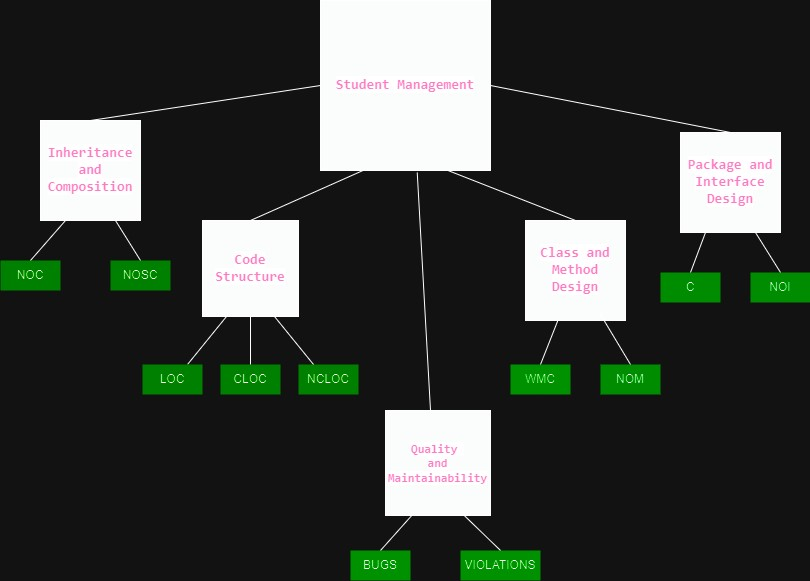


Figure 1 Structura piramidei pe baza metricilor pentru a reflecta calitatea designului aplicatiei

**Structura este formată de următoarele elemente:**

**Fundația – Code Structure:**

* LOC ( lines of code ),CLOC ( lines of comment )și NCLOC ( lines of non-comment ) – aceste valori reprezintă dimensiunea generală, lizibilitatea și structura codului.

**Baza – Class and Mehod Design:**

* NOM ( number of methods ) – numarul de metode indica complexitatea fiecarei clase.
* WMC ( weighted method complexity ) – complexitatea ponderată a metodelor asigură faptul că metodele nu devin prea complicate/complexe.

**Mijloc – Inheritance and Composition:**

* NOC ( number of direct subclasses of each class that occur in the project ) – Absența subclaselor directe sugerează o ierarhie mai simplă.
* NOSC ( number of static classes ) – Absența claselor statice indică un design mai dinamic și mai flexibil.

**Nivelul superior – Package and Interface Design:**

* C ( number of classes in each package ) – Este necesar un echilibru pentru a evita pachetele prea complexe sau simpliste.
* NOI ( number of interfaces ) – Interfețele ar putea spori flexibilitatea Codului.

**Calitate și menținere:**

* BUGS ( average bugs per class ): Media scăzută a bugurilor pe clasă este pozitivă reflectând un anumit nivel de calitate a codului.
* VIOLATIONS ( problems per class / errors or warnings ): Numărul de avertismente, erori indică domenii potențiale de îmbunătățire în ceea ce privește calitatea codului și mentenabilitatea sa.

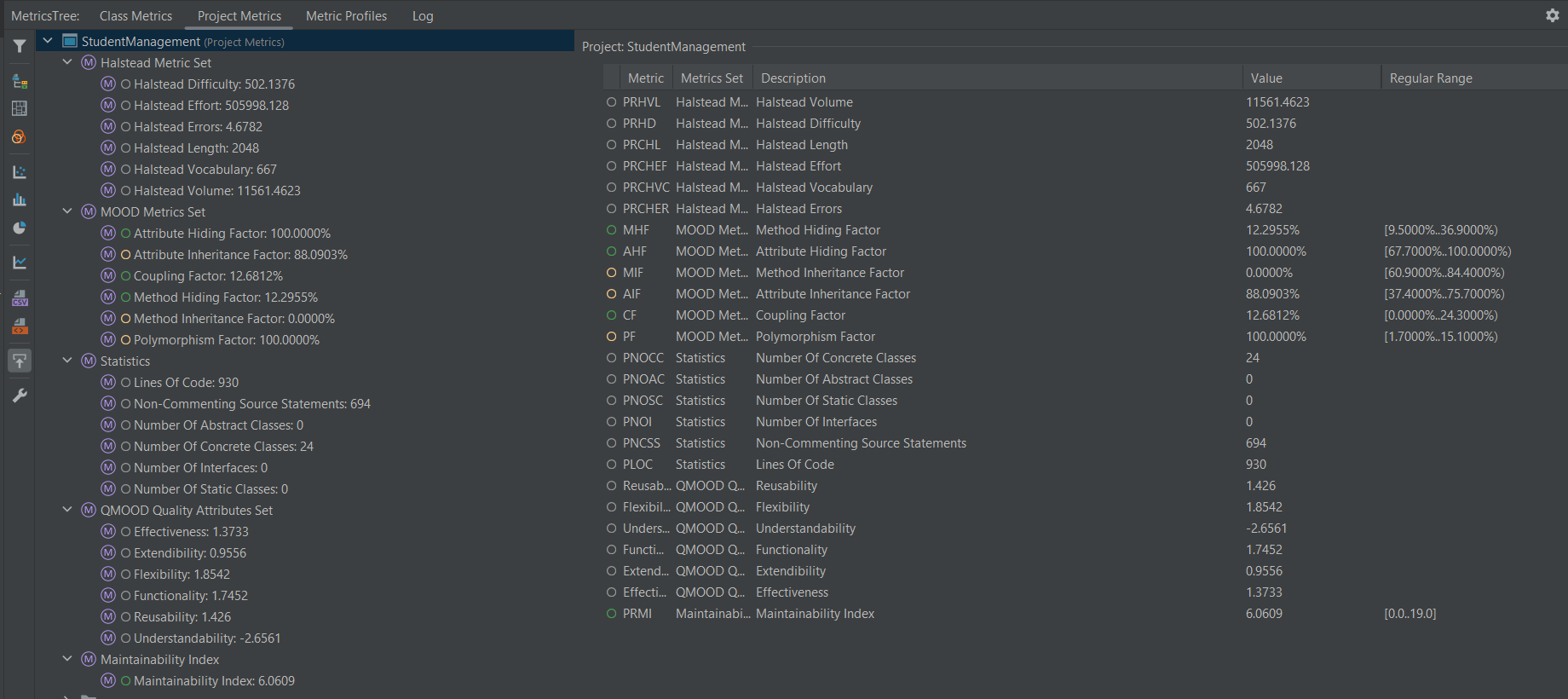


Figure 2 Valoarea metricilor obtinute cu MetricsTree

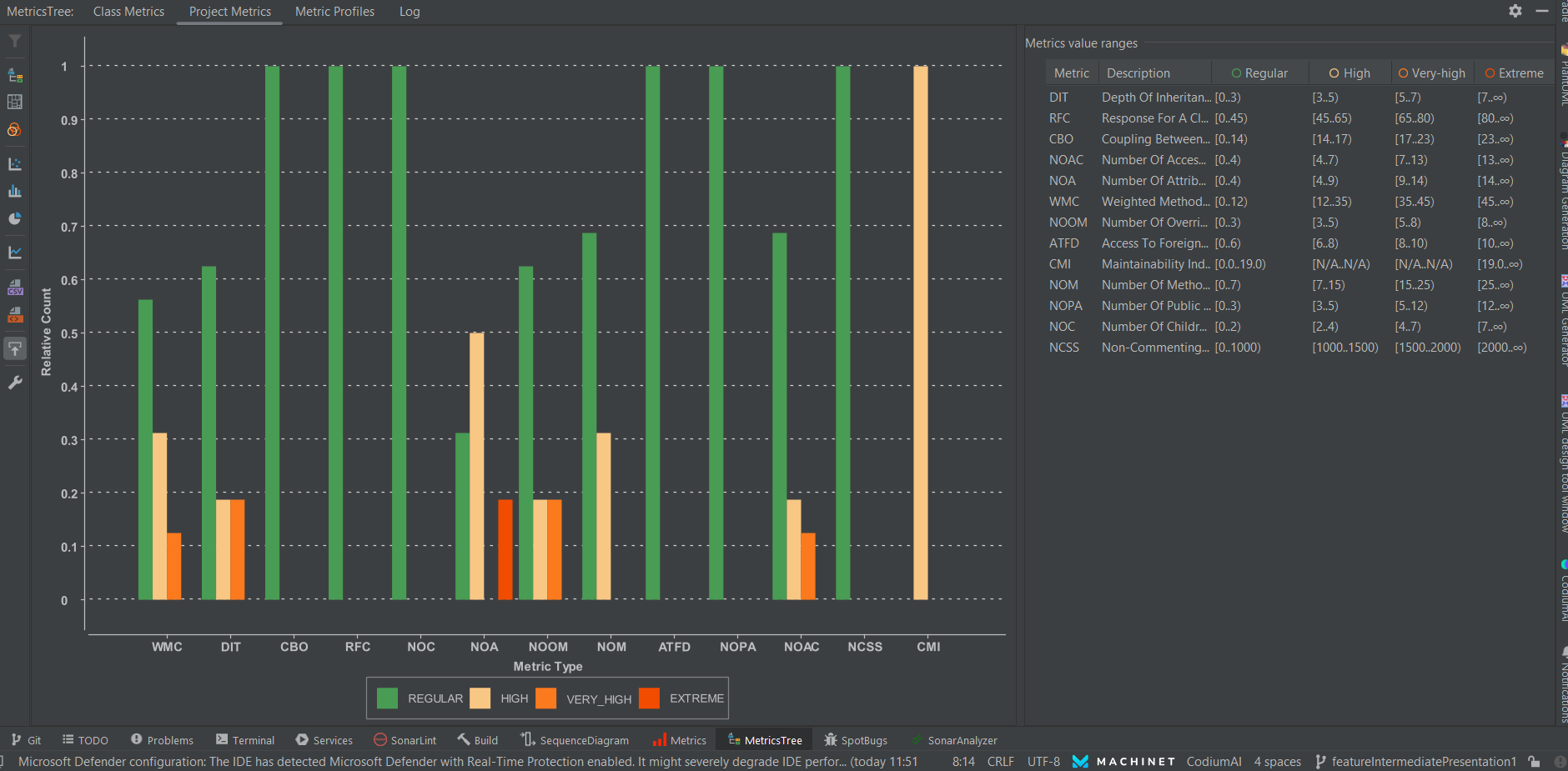


Figure 3 Valorea metricilor distribuita sub formă de chart

**Concluzia:**

* Această piramidă sugerează că aplicația dezvoltată are o bază solidă în ceea ce privește structura codului.
* Nivelurile de mijloc indică o abordare a moștenirii și compoziției.
* Nivelurile superioare necesită o privire mai atentă, mai ales in ceea ce privește designul pachetului și al interfeței.
* Punctul de vârf evidențiază domenii de îmbunătățire, concentrându-se pe reducerea erorilor și soluționarea problemelor de calitate a codului.

**References:**

**Tool used to create the previous diagram:** <https://app.diagrams.net/>