**Open Universiteit, Software Evolution (IM0202)**

**Opdracht 1: Software metrieken**

## Studentgegevens

Marco Huijben: 838316640

Ivo Willemsen:

**Samenwerking**

Wij hebben besloten om de implementatie van de metrieken te verdelen. Ivo heeft de volume, duplication en de unit test coverage metrieken uitgewerkt. Ook heeft hij de GitHub omgeving opgezet en een eerste opzet gemaakt voor de structuur van de Rascal implementatie. De metrieken voor de complexiteit voor elke unit en de size van elke unit zijn door Marco geïmplementeerd. Marco heeft ook de opzet van het verslag gemaakt.

Marco en Ivo hebben regelmatig via mail, WhatsApp en telefonisch overlegd. Tijdens deze contactmomenten is o.a. besproken op welke wijze de metrieken precies geïmplanteerd moeten worden.

**Aannames**

De volgende aannames zijn gemaakt:

1. Voor het bereken van de metriek “Complexity per unit” en “size of each unit” wordt een methode in een Java klasse als unit beschouwd. Immers een methode is de kleinst mogelijke uitvoerbare onderdeel.
2. Voor het bepalen van de waarde voor de metriek “size of each unit” is gebruik gemaakt van tabel IIIa in het artikel “Benchmark based Aggregation of Metric to Ratings”[[1]](#endnote-1).

**Resultaten**

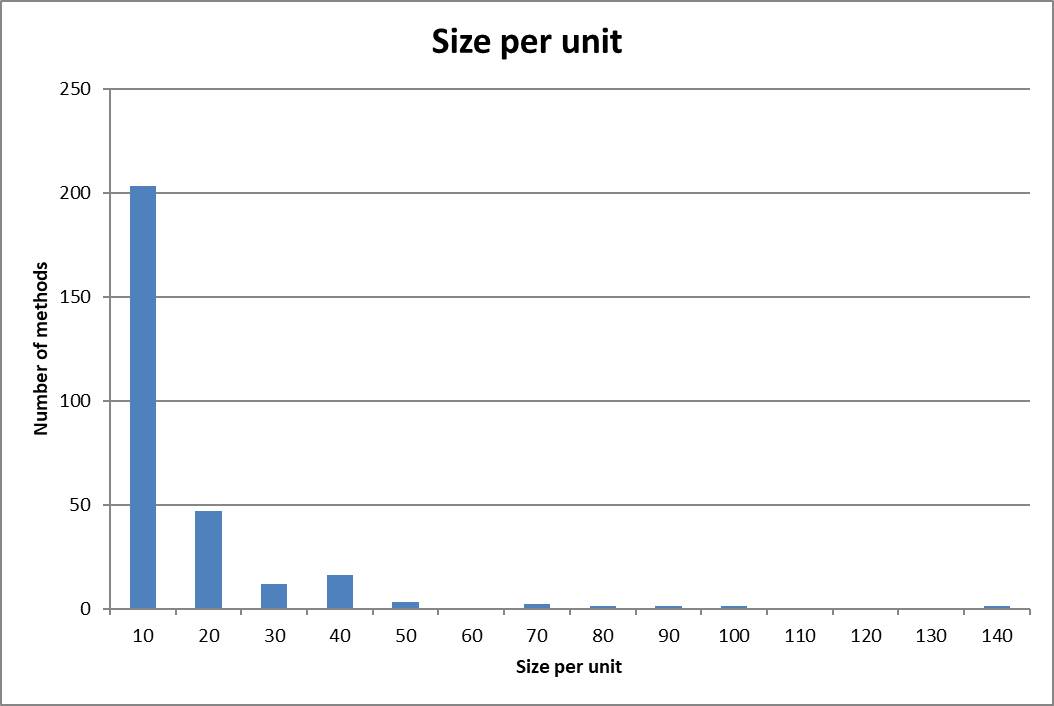
De paragraaf bevat de metrieken van de applicaties SmallSQL en HyperSQL.

**Resultaten SmallSQL**

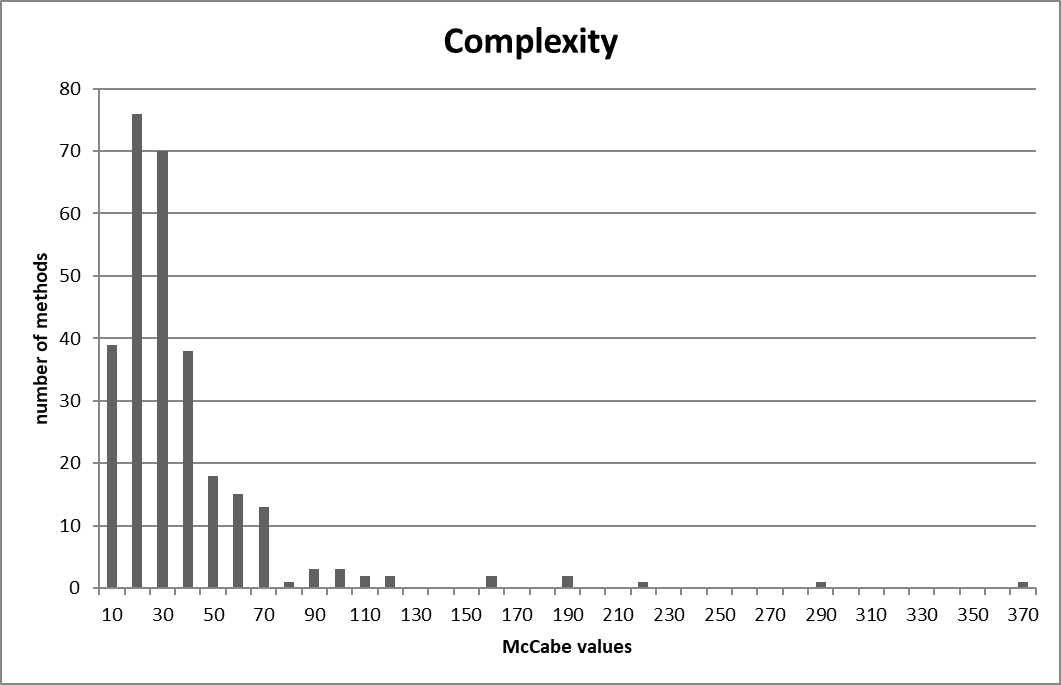
De resultaten van SmallSQL zijn:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Volume | Complexity per unit | Duplication | Unit size | Unit testing |  |
| ++ | -- | - | - | -- |
| Analysability | x |  | x | x | x | -\0 | |
| Changeablity |  | x | x |  |  | -- | |
| Stability |  |  |  |  | x | -- | |
| Testability |  | x |  | x | x | -- | |

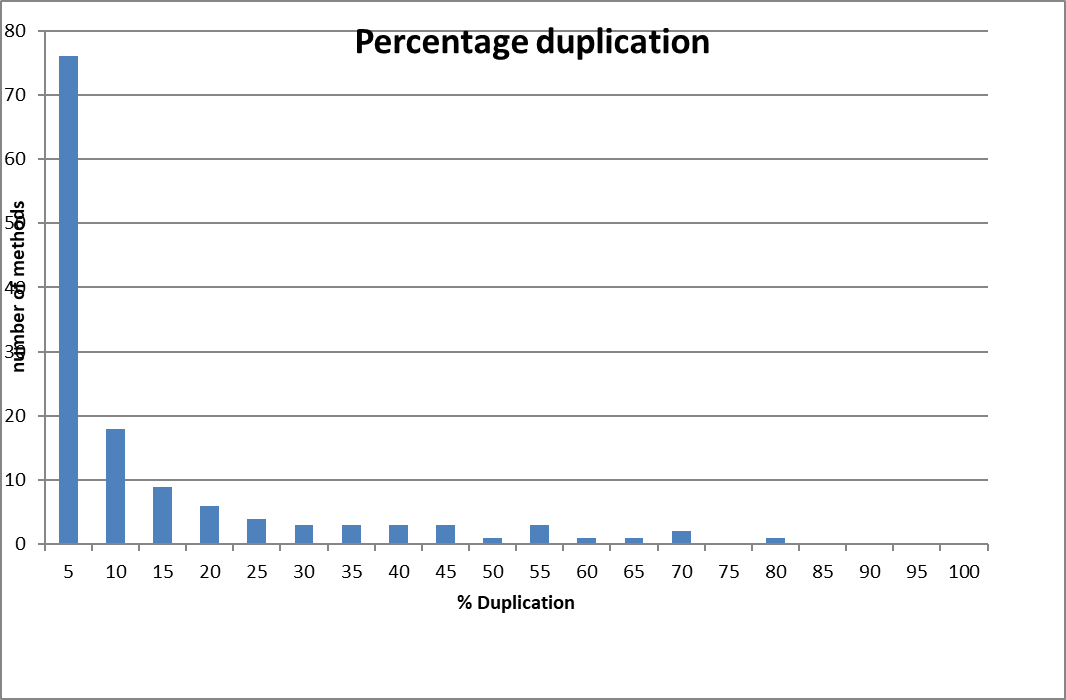
Tabel 1: SIG mapping voor SmallSQL



Figuur 1: grafiek met aantal methoden per blok unitgrootte voor smallsql



Figuur 2: Aantel methodes t.o.v. een blok McCabe waarden voor smallqsl

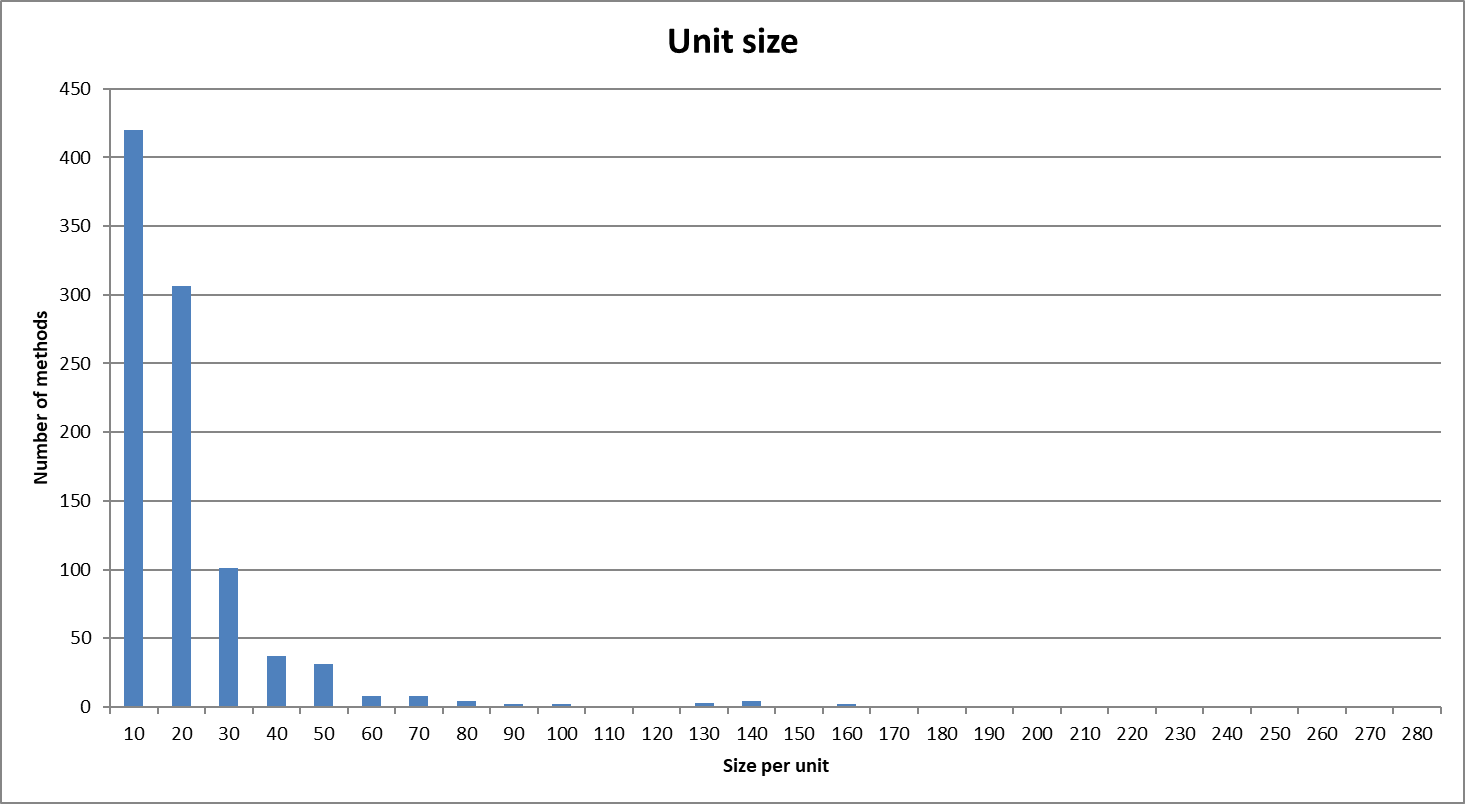


Figuur 3: Aantal methoden per duplicatieblok voor smallsql

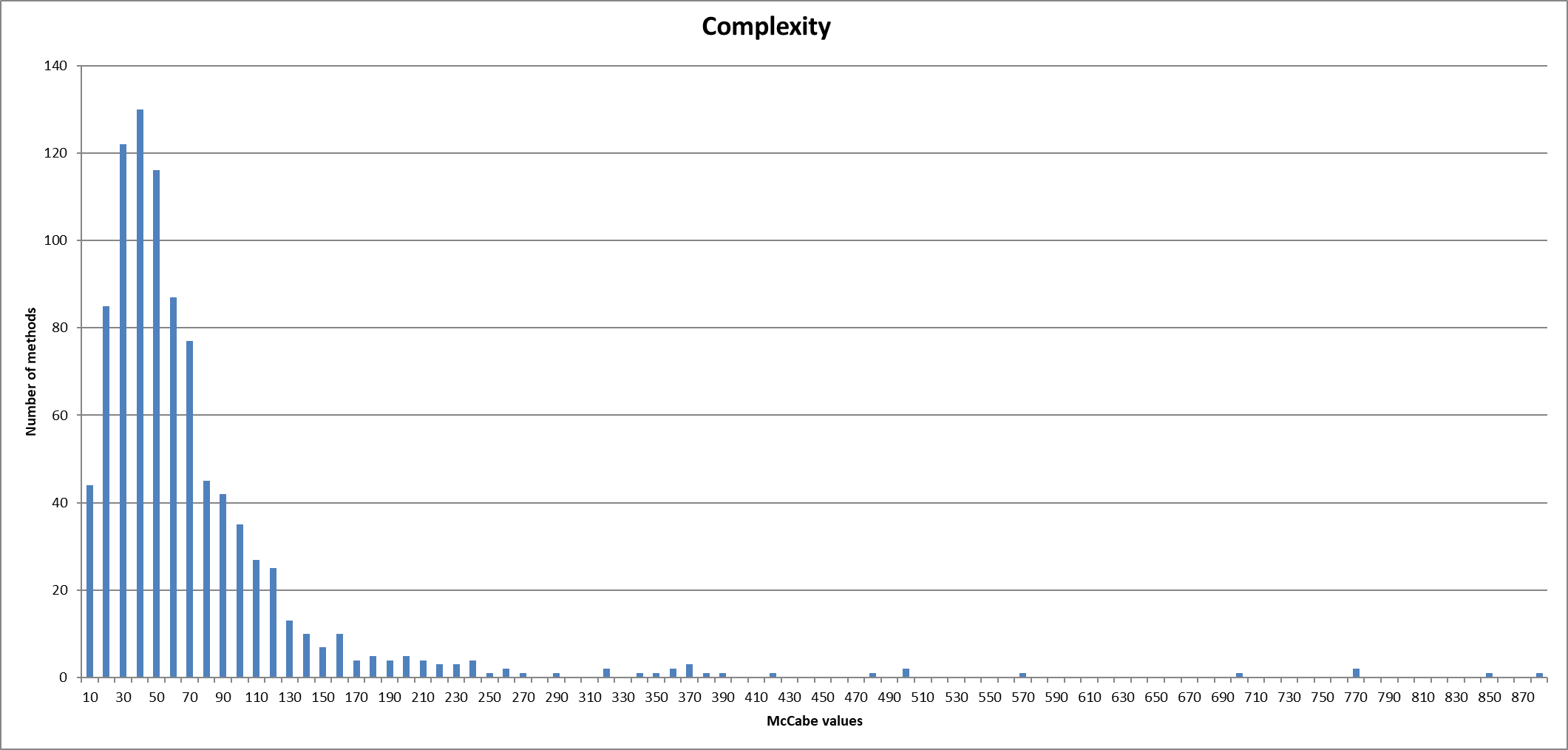
**Resultaten HyperSQL**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Volume | Complexity per unit | Duplication | Unit size | Unit testing |  |
| + | -- | - | -- | -- |
| Analysability | X |  | x | x | x | - | |
| Changeablity |  | x | x |  |  | -/-- | |
| Stability |  |  |  |  | x | -- | |
| Testability |  | x |  | x | x | -- | |

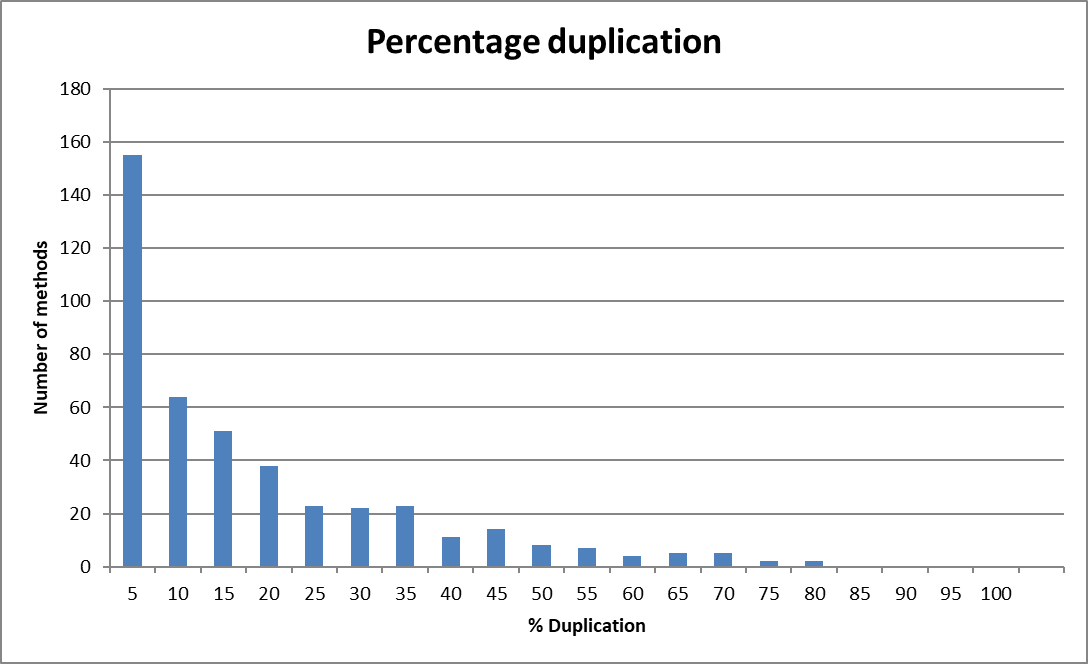
Tabel 2: SIG mapping voor HyperSQL



Figuur 4: grafiek met aantal methoden per blok unitgrootte voor hsqgldb



Figuur 5: Aantel methodes t.o.v. een blok McCabe waarden voor hsqldb



Figuur 6: Aantal methoden per duplicatieblok voor hsqldb

**Validatie**

De berekening van de complexiteit van een unit is getest door middel van een test Java applicatie. Deze applicatie bevat diverse methoden met verschillende flow control elementen ( b.v. if, while, case). Per methode is aangegeven wat de verwachte complexiteit is. Door de metrieken van de test applicatie te berekenen kan bepaald worden of de Rascal metriek berekening correct is. Op analoge wijze wordt de unit size getest.

**Interpretatie**

Applicatie SmallSql

De applicatie smallSql scoort slecht op alle vier de subcategorieën, waarbij de analysaibity het “beste” scoort. De complexiteit alsmede het aantal regels van te veel methoden is te hoog. Ook zijn er te weinig unittesten. Hierdoor is het lastig om wijzigingen door te voeren. Het aantal unit testen zal uitgebreid moet worden en de complexiteit van een aantal methoden zal moeten worden verminderd. Ook zal een refactoringslag gedaan kunnen worden, zodat de unit size van te grote methode verlaagd wordt.

**Bijlagen**

1. Een logbestand (Metrics\_smallsqk.txt) met de output van de berekeningen van de metrieken voor de applicatie smallsql.

1. https://www.sig.eu/files/en/080\_Benchmark-based\_Aggregation\_of\_Metrics\_to\_Ratings.pdf [↑](#endnote-ref-1)