Bevindingen Thijs: inCode

* pyramide geeft bij sonar core lange en smalle hierarchie, en methodes roepen veel andere methodes op. Op 480 klassen 1 god, 42 data, en 2 schizofrene, en ne hele hoop feature envy methodes
* sonar batch geeft op 381 klassen enkel 4 data klassen als probleem
* plugin api 1 data 2 schizofrene en 3 data clump methodes (lange parameterlijst)
* client: 23 data klasses, 24 feature envy methodes op 247 klasses

==> allemaal vrij proper

Bevindingen Simon:

* ik heb een programma geinstalleerd en ik heb ne hele hoop output maar ik weet niet wat ik er mee moet doen.

Bevindingen Wout:

* code is bug free. Yay.

Verbinding tussen runner en sonar: ergens w een batch aangemaakt die alle properties meekrijgt, batch roept dingen op uit plugin en da begint wss te zoeken en te scanne. Resultaten van het scannen worden op de sql server geplaatst. Widgets tonen verwerkte data. Data wordt verwerkt door sensors enzo. Thijs vraagt zich af wat er net wordt opgeslagen in de database. Alles wordt in de sql server opgeslagen.

In sonar-core zit waarschijnlijk iets om shit uit de database te halen. Er is functionaliteit voor querys.

Batch doet ook heel veel. Wss verantwoordelijk voor het analyseren zelf.

Sensor parst files en verbindt met webserver, maakt measures op laagste niveau

in de runner zit er nergens code die effectief over de code loopt. Dit gebeurt in batch? Runner doet vooral de output naar de command line, en processt user input. Maakt verbinding met de server. Runner start het runnen wel, maar gebeurt ergens anders. Alles zit in SQL database (?), runner houdt niets bij, proces wordt helemaal afgesloten. Runnen en analyse zijn niet parallel.