1. Code Coverage - какая часть кода покрыта тестами.   
   Реально это реализовать только для конкретных языков и библиотек.

Для реализации нужен функционал:

* Определения, используется ли конкретная библиотека
* Вычисления Code Coverage

При этом больше смысла рассматривать в процентах.  
Также возможно стоит делать много снапшотов, чтобы проверить, что codecoverage не уменьшается со временем.

Инструменты:

1. Codecov
2. Coveralls
3. Code Climate
4. Использование линтеров и форматтеров - по поиску конкретных имён/полей файлов-конфигов.
5. В какой день недели (спринта) чаще всего происходят коммиты. Чтобы вылавливать тех, кто начинает всё делать за день до конца спринта
6. Среднее время на закрытие пулл-реквеста. Большое время может означать:

* Плохо налажен процесс одобрения (слишком долго набирается нужное число апрувов)
* Создатели PR часто пытаются залить в мастер недоделанный код, проблемы которого выясняются только на code review перед слиянием

1. Число незакрытых ишью / Общее число ишью - больше всего смысла в рассмотрении в динамике: как сильно растёт/падает процент нерешённых задач
2. Code Churn - как часто меняется конкретный кусок кода
3. Code Ownership - пай-чарт, показывающий процент кода репозитория, написанный каждым из участников
4. Learning & growth - насколько разработчик стал лучше писать код по ходу проекта. Для реализации нужно средство для оценки качества кода (например, CodeQL).
5. PR Approval Rate