

ОТОБРАЖЕНИЕ ИЗМЕНЯЕМОСТИ МЕТОДА НА ОСНОВЕ ИСТОРИЧЕСКОЙ ИНФОРМАЦИИ В INTELLIJ IDEA

Сергей Свитков

группа 444

руководитель: к.т.н., доц., Т.А. Брыксин

рецензент: аналитик ООО "Интеллиджей Лабс", Н. И. Поваров

29 мая 2019 г.

СПбГУ

Введение

- Несколько ошибок часто находятся в одном фрагменте кода
 - *D. Wang et al, "Detect related bugs from source code using bug information", IEEE, 2010*
- Успешное исправление зачастую происходит не с первой попытки
 - *Z. Yin et al, "How do fixes become bugs?", ACM, 2011*
- Могут быть технические, архитектурные или внешние проблемы проекта
- Нужно привлечь внимание разработчика к таким фрагментам кода

Постановка задачи

Целью работы является разработка плагина для IntelliJ Platform, позволяющего визуализировать историю изменений кода

- Реализовать плагин, интегрирующийся с SDK IntelliJ Platform
- Интегрировать решение с инструментами для отслеживания рефакторингов
- Оптимизировать решение для работы с большими репозиториями

Ближайший аналог

code_call_lens: Raising the Developer Awareness of Critical Code

- Расширение VSCode для отслеживания количества вызовов каждого метода
- Добавление журналирующей компоненты
- Данные теряют актуальность при рефакторингах кода
- Похожая идея, другой подход

code_call_lens: UI

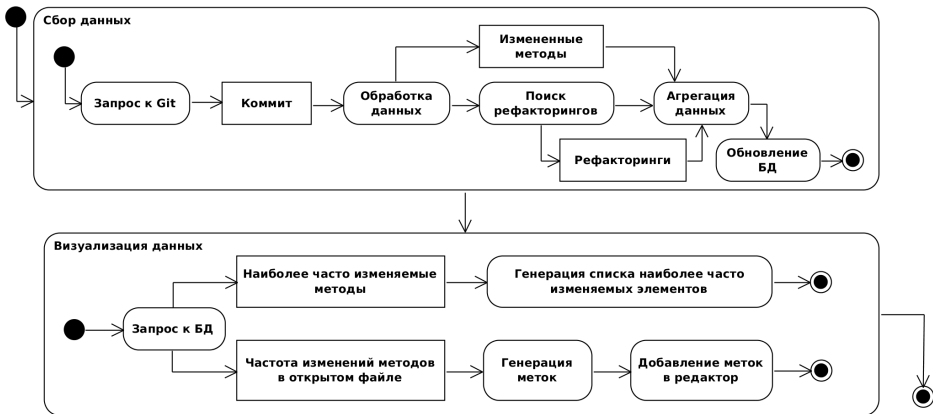
250 calls in the last 30 days

```
defp authenticate(conn, _params) do
  if conn.assigns.is_authenticated do
    conn
  else
    conn
    ▷ redirect(to: page_path(conn, :mes
    ▷ halt()
  end
end
```

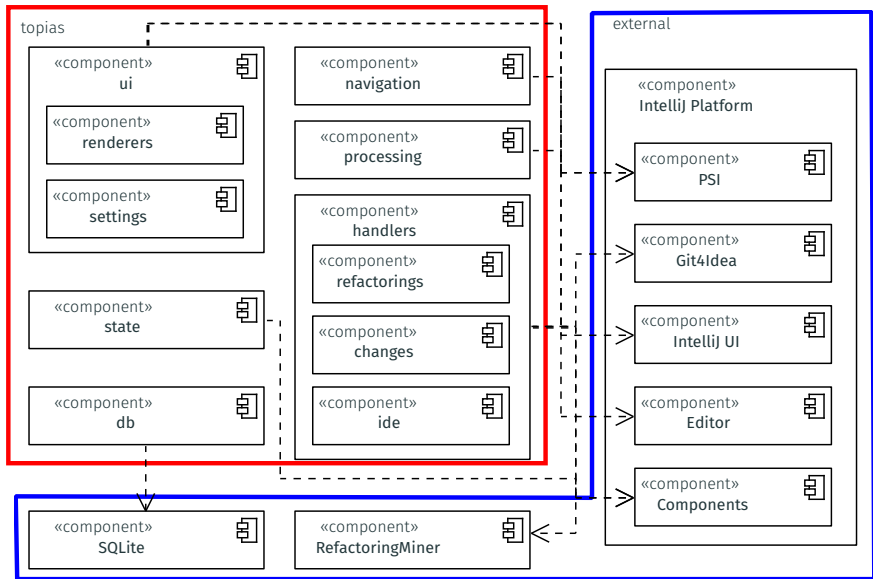
RefactoringMiner

- *Nikolaos Tsantalis et al, "Accurate and Efficient Refactoring Detection in Commit History", 40th International Conference on Software Engineering (ICSE), 2018*
- Java, Git
- Наиболее высокая точность среди существующих инструментов
- Алгоритм работы:
 - Код разбирается парсером
 - Сопоставление имен (сверху вниз)
 - Поиск рефакторингов (снизу вверх)
- Overall precision 98%, recall 87%

Предлагаемое решение



Архитектура



База данных

- Журнал изменений
- Словарь сигнатур методов
- Статистика о количестве изменений за временной интервал
 - Подсчёт статистики делегирован СУБД
 - Графики частоты изменений
 - Нормализация данных

Визуализация собранных данных

неизменяемые метки с информацией

28 Method was changed 8 time(s) for last 30 days



```
29 public List<Integer> selectChangesCountDaily(String fullSigName, DiscrType period) {  
30     final int days = period.equals(DiscrType.MONTH) ? 30 : 7;  
31     final LocalDate to = LocalDate.now();  
32     final LocalDate from = to.minusDays(days);  
33  
34  
35     final String sql = "select dtDateTime, sum(changesCount) from statisticsView where\n" +  
36         "fullSignature = ? " +  
37         "and dtDateTime between ? and ?" +  
38         "group by dtDateTime";  
39
```

Визуализация собранных данных

список ссылок на наиболее часто изменяемые методы проекта

Topias

Most changed methods for last 30 days

- m db.dao.StatisticsViewDAO.getStatDataForFile(String,DiscrType): 9 time(s)
- m db.dao.StatisticsViewDAO.selectChangesCountDaily(String,DiscrType): 8 time(s)
- m processing.CommitProcessor.processNewCommit(Collection<Change>,String,String,long): 8 time(s)
- m db.dao.MethodsChangelogDAO.insertMethodsChanges(List<MethodChangeLogEntity>): 6 time(s)
- m db.dao.MethodsDictionaryDAO.buildChangelogs(List<MethodInfo>): 6 time(s)
- m db.dao.MethodsDictionaryDAO.addToDictionary(List<MethodDictionaryEntity>): 5 time(s)
- m db.dao.MethodsDictionaryDAO.removeFromDictionary(String): 5 time(s)
- m processing.CommitProcessor.processCommit(GitCommit): 5 time(s)
- m db.DatabaseInitialization.createNewDatabase(String): 4 time(s)
- m db.dao.MethodsDictionaryDAO.upsertOfNotChangedMethodEntries(List<MethodDictionaryEntity>): 4 time(s)

6: TODO Terminal 9: Version Control Topias

Результаты

- Реализован плагин для сбора и визуализации статистики о частоте изменений методов в системе контроля версий
 - <https://github.com/ml-in-programming/topias>
 - <https://github.com/ml-in-programming/topias/releases/tag/v1.0-beta>
- Собранный статистика отображается в виде текстовых меток и графиков частоты изменений
- Доклад на конференции SEIM-2019