Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

«Южно-Уральский государственный университет»

(национальный исследовательский университет)

Институт естественных и точных наук

Факультет математики, механики и компьютерных технологий

Кафедра прикладной математики и программирования

|  |  |
| --- | --- |
|  |  |

Скрипт для просмотра игровой информации

Протокол тестирования к проекту

по дисциплине

«Технологии и модели управления проектами в программных системах»

ЮУрГУ–01.03.02.2019.78.

|  |  |
| --- | --- |
|  | Руководитель работы,  \_\_\_\_\_\_\_\_\_\_/С.М. Елсаков  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2019 г. |
|  | Автор работы  Студенты группы ЕТ-413  \_\_\_\_\_\_\_\_\_\_\_\_\_/Е.А. Велисевич  \_\_\_\_\_\_\_\_\_\_\_\_\_/И.Э. Абдулина  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_2019 г. |
|  |  |

Челябинск 2019

1. Объект испытаний

Скрипт для просмотра игровой информации о пользователе. Применяется пользователем в браузере на отдельной странице. Обозначается страницей в браузере.

1. Цель испытаний

- Выявить недочеты и недостатки в процессе эксплуатации скрипта.

- Проверка работоспособности системы;

- Проверка соответствия системы с требованиями указанными в техническом заданий;

- Проверка быстродействия системы

3. Требования к программе  
Для проведения тестов, необходимы:

- интерпретатор PHP версии не ниже чем 7.2;

- установленный и настроенный composer, для проверки зависимостей проекта;

- пакет для юнит тестирования PHPUnit;

- пакет для метрики PHPMetrics.

4. Требование к программной документации  
Техническое задание ГОСТ 34.602-89 и руководство пользователя.

5. Средства и порядок испытаний  
- Модульное тестирование покрытия кода при помощи PHPUnit;

- Нагрузочное тестирование при помощи Siege;

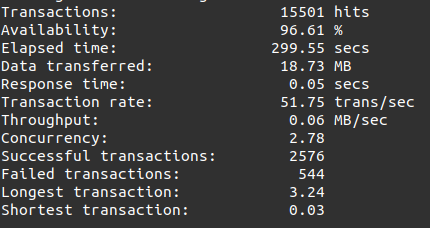
- Генерация метрик кода при помощи PHPMetrics;

6. Методы испытаний

Нагрузочное тестирование выполнялось при помощи утилиты Siege, на операционной системе Ubuntu 19.10. Модульное тестирование покрытия кода при помощи PHPUnit. Генерация метрик кода при помощи PHPMetrics.

7. Примеры, графики

Результаты следующие:



Были установлены настройки:

method=GET // method request

parser=true // parsing response

protocol=HTTP/2 // Protocol for ssl connecting

limit=50 // threads

connection=keep-alive // connection-directive

concurrent=25 // count simulated users

internet=true // random url

time=5M // time for testing

delay=1 // delay per requests

timeout=5 // timeout connect

Из выявленных недостатков: слабая пропускная способность интернет-канала, которая не позволяет пропускать запросы в более чем 50 потоков.

- Покрытие кода;

= Результат: --

- Нагрузочное тестирование:

= Результат:

Кол-во запросов - 15501

Затраченное время - 299.55 секунд

Среднее время ответа - 0.05сек

Доступность - 96.61%

Успешных запросов(200 код) - 2576

Ошибок(не 200 код) - 544

Данных передано - 18.44мб

- Генерация метрик кода:

= Результат:

Maintainability index(качество кода):

src\Script\Data - 105.93

src\Script\Script - 50.55

src\Script\Game - 88.45

src\Script\MyFormatter - 82.72

src\Script\ViewData - 93.71

src\Script\Ajax - 95.54

Violations(нарушения) - 4

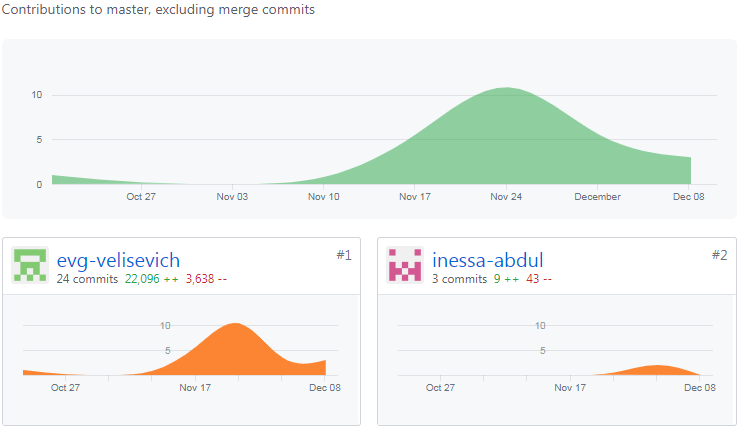
Lines of code(строк кода) - 1135

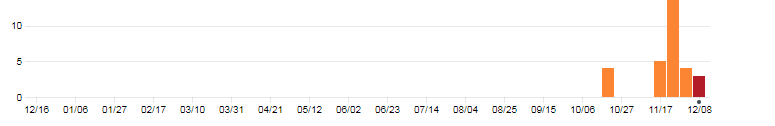
Classes(количество классов) - 6

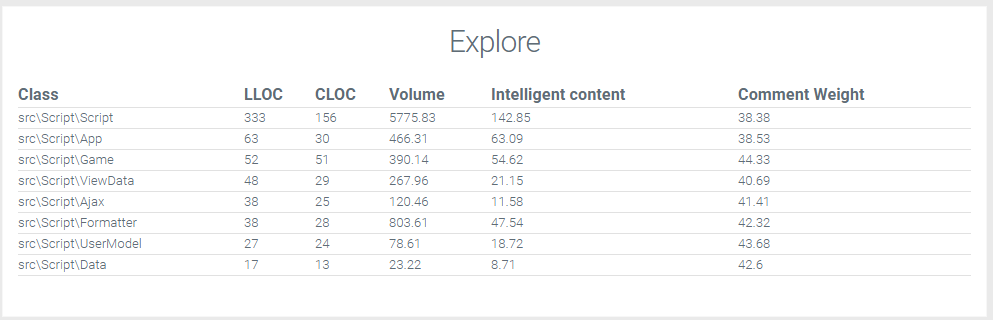
Average cyclomatic complexity by class(среднее значение цикломатической сложности на класс) - 11.5

Assertions in tests - —

Average bugs by class(среднее значение багов на класс) - 0.43





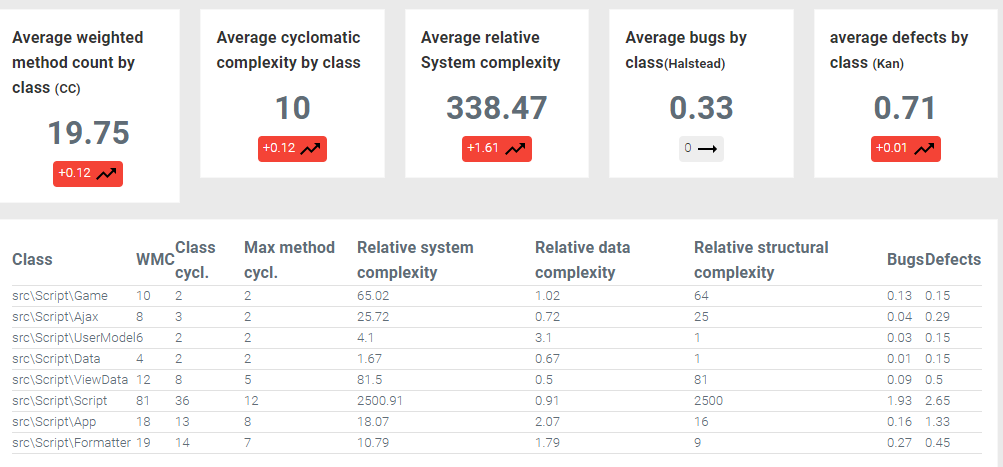


LLOC(количество логических строк кода)

Volume(объем)

Intelligent content(процент строк кода)

Comment weight(процент откомментированных строк кода)



Average weighted method count by class (CC)(Средневзвешенное количество методов по классам)

Average cyclomatic complexity by class(Средняя цикломатическая сложность по классу)

Average relative System complexity(Средняя относительная сложность системы)

Average bugs by class(Halstead)(Средние ошибки по классу)

Average defects by class (Kan)(Средние дефекты по классу)

WMC

Class cycl.

Max method cycl.

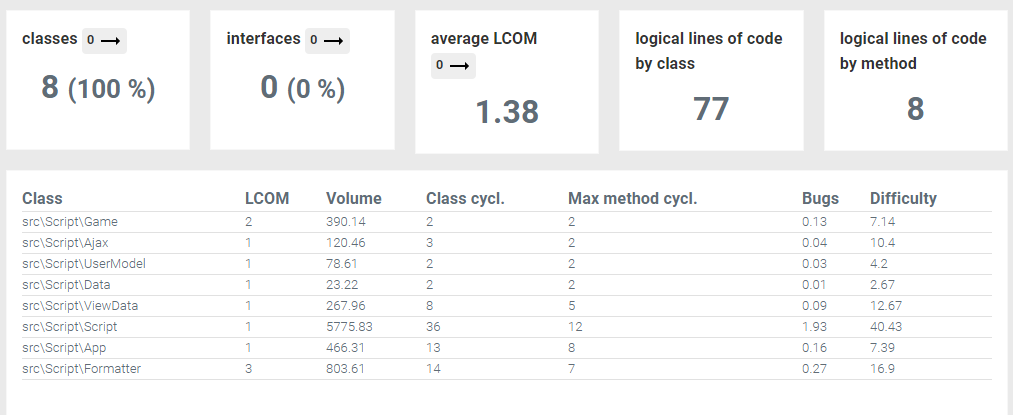
Relative System complexity(Относительная сложность системы)

Relative Data complexity(Относительная сложность данных)

Relative structural complexity(Относительная структурная сложность)

Bugs(Ошибки)

Defects(Дефекты)



logical lines of code method(кол-во логических строк метода кода)

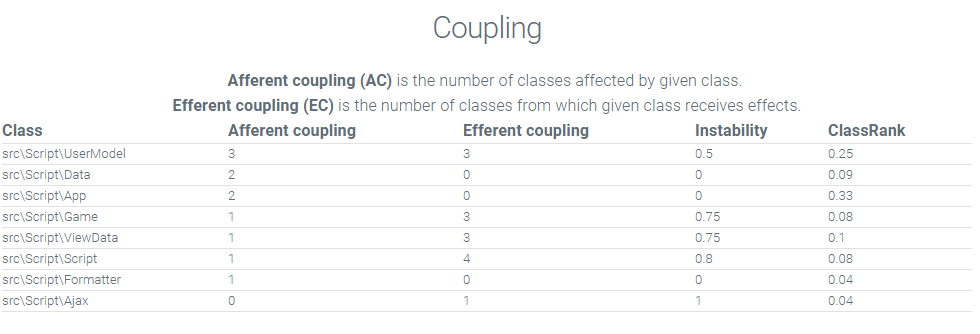
logical lines of code by class(кол-во логических строк класса)

Average LCOM(среднее значение количества логических строк кода)

Interfaces(кол-во интерфейсов)

Difficulty(сложность кода)

Coupling (связь)



Afferent coupling(афферентная связь)

Efferent coupling(эфферентная связь)

Instability(нестабильность)

ClassRank(ранк класса)