|  |  |
| --- | --- |
| Пермский филиал федерального государственного автономного образовательного учреждения высшего образования  «Национальный исследовательский университет  «Высшая школа экономики» | |
| *Факультет экономики, менеджмента и бизнес-информатики* | |
| Коковин Алексей Николаевич  Иванов Михаил Викторович | |
| Constructive Cost Model | |
| *Практическая работа* | |
| по направлению подготовки *09.03.04 Программная инженерия*  образовательная программа «Программная инженерия» | |
|  | Преподаватель кафедры информационных технологий в бизнесе \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  В. П. Коротун |

|  |
| --- |
| Пермь 2019 |

# FP to KLOC

Задача уже была разбита на функциональные точки, но расчёт был сделан для задачи в целом. Для данной задачи необходимо провести вычисления заново, но уже не для в целом, а для каждого компонента индивидуально. Результат этих расчётов представлен на таблице ниже.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Компонент** | **UFP** | **TDI** | **VAF** | **AFP** |
| **Получения контента** | 10 | 16 | 0,81 | 8,1 |
| **Монетизация контента** | 7 | 14 | 0,79 | 5,53 |
| **Модерация контента** | 80 | 22 | 0,87 | 69,6 |
| **Распространение контента** | 10 | 22 | 0,87 | 8,7 |

Далее нужно определить количество тысяч строк кода каждого из компонентов в зависимости от AFP.

[coef source](https://www.qsm.com/resources/function-point-languages-table)

С коэффициентами, взятыми по ссылке, был проведён расчёт и получен результат, показывающий что количество строк кода для большинства компонентов не превышает тысячи.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Получения контента** | **Монетизация контента** | **Модерация контента** | **Распространение контента** |
| **AFP** | 8,1 | 8,7 | 5,53 | 69,6 |
| **Коэффициенты языков:** | <https://www.qsm.com/resources/function-point-languages-table> | | |  |
| **Язык1** | js | sql | sql | sql |
| **Коэф языка 1** | 47 | 21 | 21 | 21 |
| **Язык2** | html | python | python | python |
| **Коэф языка 2** | 34 | 47 | 47 | 47 |
| **Доля языка 1** | 0,7 | 0,2 | 0,2 | 0,1 |
| **Итого KLOC** | 0,34911 | 0,36366 | 0,231154 | 3,09024 |

\*python ≈ javascript

Вместо коэффициента для питона был взят коэффициент для javascript, так как языки похожи по синтаксису и по набору библиотек для анализа данных.

# SF

## PREC

Команда разработчиков будет набрана скорее всего из студентов, поэтому опыт работы над проектами будет меньше желаемого.

Experience: Low 3,72

## FLEX

Процесс разработки остаётся очень гибким, поскольку ни у кого нет опыта работы в команде, из чего следует отсутствие, к примеру, общепринятого стиля кода или одинакового тулчейна (toolchain).

Гибкость процесса разработки: High 2,03

## RESL

Риски данного проекта плохо отработаны, так как нету опыта работы над проектами, из чего следует отсутствие представления о том, что может являться риском.

Отработанность рисков: Very Low 7,07

## TEAM

Опять же, по причине малого опыта команды и способа её набора опыт работы и налаженность данной конкретной команды не высока.

"Налаженность" команды: Low 4,38

## PMAT

Поскольку команда раньше не работала в таком составе, ни на проектах, процессы взаимодействия между членами команды не налажены окончательно.

Зрелость процессов: Low-Nominal 6,24−4,68

# EM

|  | **Получения контента** | **Монетизация контента** | **Модерация контента** | **Распространение контента** |
| --- | --- | --- | --- | --- |
| параметры персонала: | | | | |
| ACAP (анализ capablity) | 1,29 | 1,29 | 1,29 | 1,29 |
| AEXP (анализ experience) | 1,0 | 1,0 | 1,1 | 1,22 |
| PCAP (programmer capability) | 1,15 - 1,0 | 1,15 - 1,0 | 1,15 - 1,0 | 1,15 - 1,0 |
| PCON (продолжительность работы персонала) | 1,15 - 1,0 | 1,15 - 1,0 | 1,15 - 1,0 | 1,15 - 1,0 |
| PEXP (опыт работы с платформой) | 1,0 | 1,0 | 1,0 | 1,0 |
| LTEX (Опыт использования языка и инструментов) | 1,09 - 1,0 | 1,09 - 1,0 | 1,09 - 1,0 | 1,09 - 1,0 |
| параметры продукта: | | | | |
| RELY (требуемая надёжность) | 1,0 | 1,10 - 1,26 | 1,10 | 1,26 |
| DATA (Размер БД) | 1,28 | 1,0 | 1,28 | 1,28 |
| CPLX (Сложность) | 1,0 | 1,0 | 0,87 | 1,17 |
| RUSE (Повторное использование) | 1,0 | 0,95 | 0,95 | 1.0 |
| DOCU (Требование к документации) | 1.0 | 1.11 | 1.11 | 1.11 |
| параметры платформы: | | | | |
| TIME (Ограничение времени работы приложения) | 1.0 | 1.11 Время-деньги :) | 1.0 | 1.29 |
| STOR (Ограничение памяти) | 1.0 | 1.0 | 1.0 | 1.0 |
| PVOL (Частота изменений) | 1.0 | 1.15 | 0.87 | 1.0 |
| параметры проекта: | | | | |
| TOOL(Используемые инструменты) | 1.09 - 1 | 1.09 - 1 | 1.09 - 1 | 1.09 - 1 |
| SITE(Способы удалённого сотрудничества) | 0,93-0,86 | 0,93-0,86 | 0,93-0,86 | 0,93-0,86 |
| SCED(Рейтинг "расширения/сжатия графиков") | 1,14-1,0 | 1,14-1,0 | 1,14-1,0 | 1,14-1,0 |

# Расчёты

## Полный размер

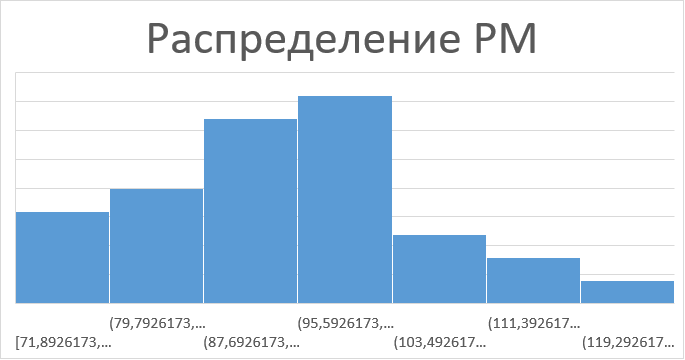
## PM для SCED

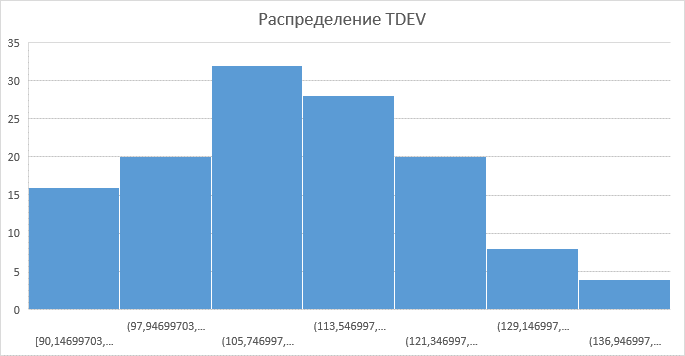
## Доля PM SCED для каждого компонента

## PM полная для каждого компонента

## PM результат

## Гистограммы распределений PM и TDEV





Как можно заметить, распределение весьма похоже на нормальное. Рассчитаем статистические величины.

|  |  |
| --- | --- |
| PM mean | 93,01128608 |
| PM дисп | 148,0880263 |
| PM ср откл | 9,500608125 |
|  |  |
| TDEV mean | 103,9233274 |
| TDEV дисп | 192,6265638 |
| TDEV ср откл | 10,89790056 |

|  |  |  |
| --- | --- | --- |
| prediction | from | to |
| PM | 83,51068 | 102,5119 |
| TDEV | 93,02543 | 114,8212 |

