

## Технический долг Kotiki team



© Bird Born

### 1. Инспектирование кода

а) Непонятный / нечитабельный код:

На данном этапе разработки проекта код понятен и читабелен. Однако, некоторые методы приложения состоят из большого числа инструкций (напр., методы классов-парсеров), что в значительной степени затрудняет понимание кода для людей, не задействованных в процессе разработки.

б) Дублирующийся код:

В некоторых модулях приложения код повторяется (в классах-парсерах).

в) Отсутствие автоматизации

Автоматизация отсутствует.

г) Запутанная архитектура и ненужные сложные зависимости:

Приложение разрабатывается с учетом архитектуры MVP. Однако, мы реализовали архитектуру с существенным недостатком: у нас слой Presenter получился очень тонким. Многие задачи слоя Presenter выполняет слой Model.

д) Незакомиченный код:

Забыли закоммитить тесты.

е) Отсутствие / несоответствие технической документации:

Документация присутствует.

ж) Отсутствие тестовой среды:

Среда есть и тесты присутствуют. Но забыли закоммитить.

з) Длинные циклы интеграции / отсутствие непрерывной интеграции:

CI отсутствует.

## **2. План мероприятий**

а) Исправить ошибки архитектуры. А именно: перенести некоторые функции слоя Model на слой Presenter.

б) На слое Model выделить общую функциональность классов-парсеров и вынести ее в общий базовый класс

в) Провести рефакторинг существующего кода. А именно: разобраться со слишком длинными методами классов-парсеров.

г) Закоммитить тесты

д) Настроить CI

## **3. Оценка плана мероприятий**

В целом план мероприятий имеет приемлемый объем, однако выполнить план во время текущего спринта (Спринт 3) будет затруднительно.

## **4. Сравнение объема технического долга (ТД) и недореализованных фич (НФ)**

ТД = 2 + 7 + 12 = 21 sp

НФ (Спринт 3) = 18 + 21 + 12 = 51 sp

НФ (Спринт 4) = 18 + 18 = 36 sp

## 5. Вывод

Технический долг необходимо устранить для облегчения поддержки проекта в будущем. Поскольку объем технического долга приблизительно равен половине объема запланированных на текущий спринт задач, то было решено некоторую часть ТД устранить в течение текущего спринта, а некоторую - в течение следующего. *На протяжении текущего спринта* команда будет устранять нарушения в MVP-архитектуре приложения. *На следующий спринт* мы оставим устранение отсутствия тестов и настройку CI.