УДК – XXX.XXX

# Кодогенерация как способ оптимизации времени в отделе мобильной разработки

Магистрант 1 курса И. В. Кислюк

Руководитель - старший преподаватель С. В. Одиночкина

*Университет ИТМО, Санкт-Петербург*

*igorkislyuk@icloud.com*

* Вступление.
  + Треугольник разработки. Важность оптимизации времени
  + Определение кодогенерации. Виды
  + Мобильный отдел в промышленности
* Основная часть
  + Способы оптимизации в мобильном отделе
  + Обзор кроссплатформенных решений
  + Кодогенерации в рамках одной платформы. Генерация картинок, строк локализации
  + Кодогенерация в рамках проекта
* Вывод.
  + Указать почему кодогенерация приемлимый способ оптимизации времени
  + Достоинства и недостатки
  + Будущее развитие кодогенерации

В современном мире при разработке ПО важным моментом становится поиск баланса между ключевыми параметрами. К ним относят требования, временные и финансовые затраты. Выигрыш в одном параметре позволяет выиграть в остальных. Например, оптимизация затрат времени команды сократит финансовые расходы. С целью сокращения затрат на разработку проектов, активно проводятся исследования в рамках оптимизации времени и большинство компаний и команд представляют свои решения тех или иных задач. Оптимизация может проводиться на различных уровнях проекта – на уровне управления, например, смена модели разработки, или на уровне работы с программным кодом, к таковым относят автоматизацию код-ревью в командах и генерацию кода.

Кодогенерация – процесс генерации кода на основе определенных данных. Различают несколько видов кодогенерации: на основе более высокоуровневого кода (использование расширений LINQ в языке высокого уровня C#), на основе метаданных (подход «Database First» в Entity Framework) и на основе шаблонов (формирование страниц ответа в серверном JavaScript «ejs»).

Современные мобильные приложения представляют из себя «тонких клиентов» в клиент-серверной архитектуре. Важным моментом является правильная организация взаимодействия с сервером и предоставление пользовательского интерфейса в правильной цветовой палитре и с правильными компонентами – изображения в строго определенных размерах и строки локализации.

Множество решений различных компаний используются для генерации программного кода цветов, строк локализации, однако не было найдено подходящее решение для оптимизации взаимодействия с сервером – необходимо было упростить реализацию взаимодействия клиента с сервером.

На основании этого были предъявлены следующие требования: возможность получения сгенерированных файлов запросов к серверу для любой платформы, готовых моделей классов / структур на любом языке программирования, а также возможность получения сгенерированной документации разработчиков. Согласно требованиям, была разработана слоистая архитектура приложения, которая разделяла платформо-независимое описание методов сервера и моделей, представленное в сырой модели, промежуточная модель, сформированная для отображения данных на файлы программного кода, используя шаблоны. На основании разработанной архитектуры приложения, удалось добиться повышения общей скорости разработки проектов. Дописать основную часть

Представленный вид кодогенерации позволяет оптимизировать временные затраты в проектах, почти устраняя дублирование реализации клиент-серверного взаимодействия на разных платформах, таких как iOS, Android, Windows Phone. Среди недостатков стоит отметить первоначальную сложность интегрирования в новые проекты, невозможность полноценного использования в существующих проектах и неопытность разработчиков при внедрение такого способа оптимизации. Кодогенерация призвана упростить разработку схожих мобильных бизнес-приложений и активно ведется разработка таких проектов, как React Native и Apache Cordova.

**ЛИТЕРАТУРА**

1. https://habrahabr.ru/post/23858/
2. https://habrahabr.ru/post/23795/
3. https://file.scirp.org/pdf/JSEA20110900003\_24019188.pdf
4. https://researcher.watson.ibm.com/researcher/files/zurich-jku/mdse-08.pdf
5. An analysis of tools for automatic software development and automatic code generation http://www.scielo.org.co/scielo.php?script=sci\_arttext&pid=S0120-62302015000400010
6. https://facebook.github.io/react-native/
7. https://cordova.apache.org