

# 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*

В разработке программного обеспечения ключевым моментом становится баланс между основными показателями – требованиями, временем и стоимостью, а также их оптимизация [2]. Кодогенерация – процесс генерации кода на основе определенных данных. Различают несколько видов кодогенерации: на основе более высокого уровня 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») [1].

Современные мобильные приложения представляют собой «тонких клиентов» в рамках клиент-серверной архитектуры, делая больший упор на отзывчивость и пользовательский интерфейс. Кодогенерация должна применяться как со стороны пользовательского интерфейса, так и со стороны клиент-серверного взаимодействия. Существует множество открытых решений для автоматизации создания пользовательского интерфейса, а также для генерации файлов мультимедиа и компонентов локализации приложений [5]. В процессе исследования не было найдено решение для генерации компонентов взаимодействия с сервером. На основе этого была выдвинута идея разработки генератора, представленного в виде самостоятельного приложения.

Для этого были предъявлены следующие требования к разрабатываемому приложению: получение моделей запроса и ответа с возможностью адаптации под любую платформу и язык программирования, а также генерация документации для разработчиков. Согласно требованиям, была спроектирована многослойная архитектура генератора, которая состоит из моделей описания запросов и ответов сервера, промежуточных моделей для реализации полиморфных интерфейсов и результирующих моделей для отображения их в файлы программного кода. По завершению разработки генератора, была проведена успешная интеграция в новые проекты компании. По результатам использования генератора на проектах, общее время реализации взаимодействия с сервером суммарно на платформах сократилось на 9-13%, что дает основание считать данную оптимизацию выигрышной.

Представленный вид кодогенерации позволяет оптимизировать временные затраты в проектах, почти устранив дублирование реализации клиент-серверного взаимодействия на различных мобильных платформах. Среди недостатков стоит отметить первоначальную сложность интегрирования в новые проекты, невозможность полноценного использования в существующих проектах. Кодогенерация призвана упростить целиком разработку мобильных бизнес-приложений. Примерами таких проектов являются React Native и Apache Cordova.

## **ЛИТЕРАТУРА**

1. Подходы к кодогенерации [Электронный ресурс]. – Режим доступа: <https://habrahabr.ru/post/23858/> свободный. Язык рус. (дата обращения 04.02.2018)
2. Research and Application of Code Automatic Generation Algorithm Based on Structured Flowchart.– Harbin Institute of Technology, 2011.– 12 с.
3. React Native [Электронный ресурс]. – Режим доступа: <https://facebook.github.io/react-native/> свободный. Язык англ. (дата обращения 04.02.2018)
4. Apache Cordova [Электронный ресурс]. – Режим доступа: <https://cordova.apache.org> свободный. Язык англ. (дата обращения 04.02.2018)
5. Генератор ресурсов интерфейса [Электронный ресурс]. – Режим доступа: <https://github.com/SwiftGen/SwiftGen> свободный. Язык англ. (дата обращения 04.02.2018)