

А. А. Подолянчик, Е. А. Блинова
(УО «БГТУ», Минск)

**РАЗРАБОТКА ПО ДЛЯ МОБИЛЬНЫХ УСТРОЙСТВ
ПОД УПРАВЛЕНИЕМ ОПЕРАЦИОННОЙ СИСТЕМЫ
ANDROID С ИСПОЛЬЗОВАНИЕМ ФРЕЙМВОРКА REACTIVEX**

Фреймворк `ReactveX` позволяет работать со всеми популярными языками объектно-ориентированного программирования. Реактивная

разработка строится на постоянном отслеживании состояния объекта. Фреймворк предоставляет ряд расширений, которые включают в себя особенности используемого языка, в частности, для ориентированных на мобильную разработку. Суть реактивного программирования заключается в том, что объект, состояние которого может измениться, является своего рода издателем и позволяет другим объектам подписываться на его события. В мире мобильной разработки объектом могут выступать любые виды данных, процессы. Существует возможность преобразования, объединения, отложенного выполнения, формирования новых источников данных, что в свою очередь позволяет производить модификации исходной информации, предоставляемой издателем. Обработка результатов публикуемых объектов, происходит благодаря методам `onNext()`, `onError()`, `onComplete()`, которые позволяют определить, успешно ли завершился процесс передачи информации, и нужно ли ожидать каких-либо других данных от издателя, что гарантирует отказоустойчивость мобильного приложения.

Для демонстрации было разработано мобильное приложение «Заметки». При написании использовался паттерн проектирования MVVM. Любое действие пользователя является сигналом к началу обработки данных. При создании или модификации заметки используется одна активность, причем, если данная активность не получает сигнала о том, что был получен объект «Заметка», поля остаются пустыми, и при нажатии кнопки "Create note" создается новая запись в локальной БД. Если же объект был получен, то текст кнопки меняется на "Modify note" и при нажатии на нее данная запись обновляется. Исходный код можно посмотреть в <https://github.com/alinalinalina123/ReactiveSample>.

Данный фреймворк решает проблему зависания интерфейса, за счет того, что вычисления можно распределить между другими потоками, что позволит пользователю взаимодействовать с приложением даже во время выполнения сложных операций в фоновых потоках.