

Разработка средств настраиваемой репликации для СУБД SQLite

АВТОР: АНФАЛОВ ЕГОР ВАДИМОВИЧ,
СТУДЕНТ ГРУППЫ КЭ-405, КАФЕДРЫ ЭВМ,
ВШ ЭКН, ЮУРГУ

РУКОВОДИТЕЛЬ ПРОЕКТА: ЯРОШ ЕЛЕНА
СЕМЁНОВНА К.Т.Н., ДОЦЕНТ

Цель

Разработка программного комплекса на базе операционной системы Android для организации передачи файлов журнала транзакций базы данных SQLite по среде передачи данных без использования хостинг сервера в качестве посредника.

Задачи

- ▶ Рассмотреть современные способы передачи данных между мобильными устройствами.
- ▶ Провести детальный анализ каждого из средств по критериям радиуса действия, скорости передачи данных и использования хостинг сервисов.
- ▶ Разработать модуль для разработчиков мобильных приложений, который осуществляет репликацию базы данных на двух мобильных устройствах.
- ▶ Разработать мобильное приложение для демонстрации функционала разработанного модуля репликации.
- ▶ Оценить работоспособность разработанного программного комплекса.

Обзор аналогов

4

Критерий	LiteSync	Разрабатываемый модуль репликации
Расширяет существующий функционал базы данных SQLite	+	+
Использует локальную базу данных при отсутствии соединения между устройствами	+	+
Передача только новых данных	+	+
Кроссплатформенность	+	-
Использует хостинг сервисы	+	-
Лицензирование	+	-

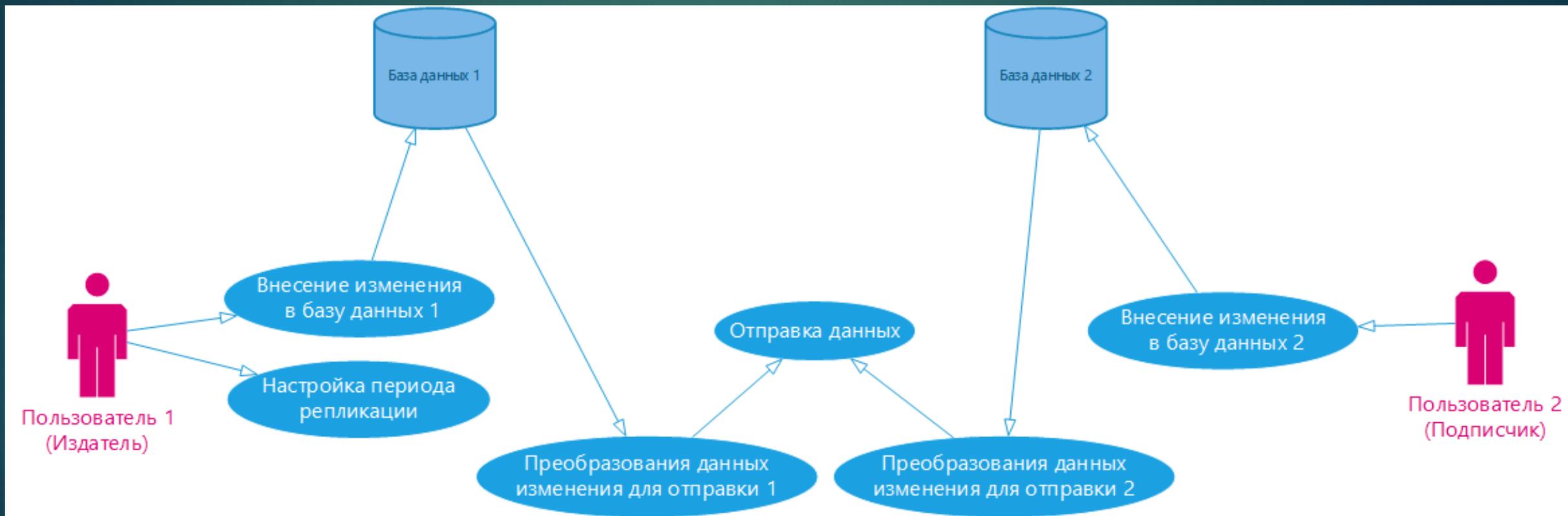
Выбор способов передачи данных

5

Способ передачи данных	Радиус действия	Скорость передачи данных	Использование хостинг сервисов
Мобильный интернет	Зона сигнала вышки связи (до 2 км)	До 1 Гбит/с	Да
Wi-Fi	До 50 м	До 1 Гбит/с	Да
Bluetooth	До 10 м	До 720 Кбит/м	Нет
NFC	До 10 см	400 Кбит/с	Нет
SMS	Зона сигнала вышки связи (до 2 км)	9,6 кбит/с	Нет

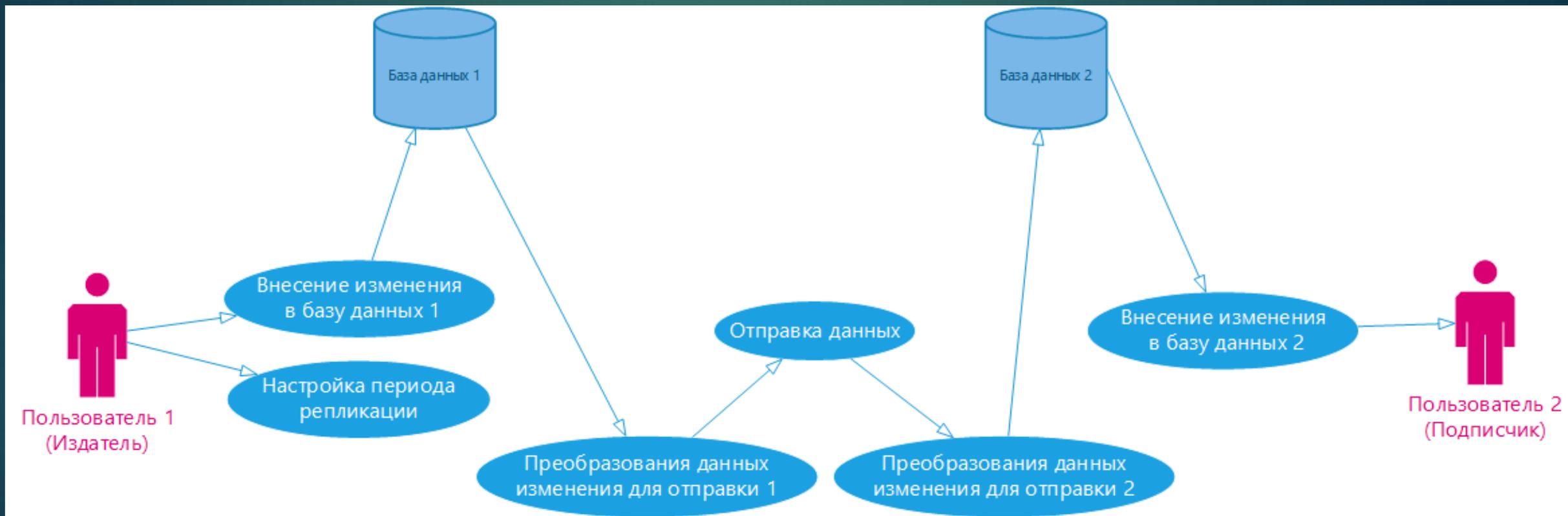
Сценарии использования

6



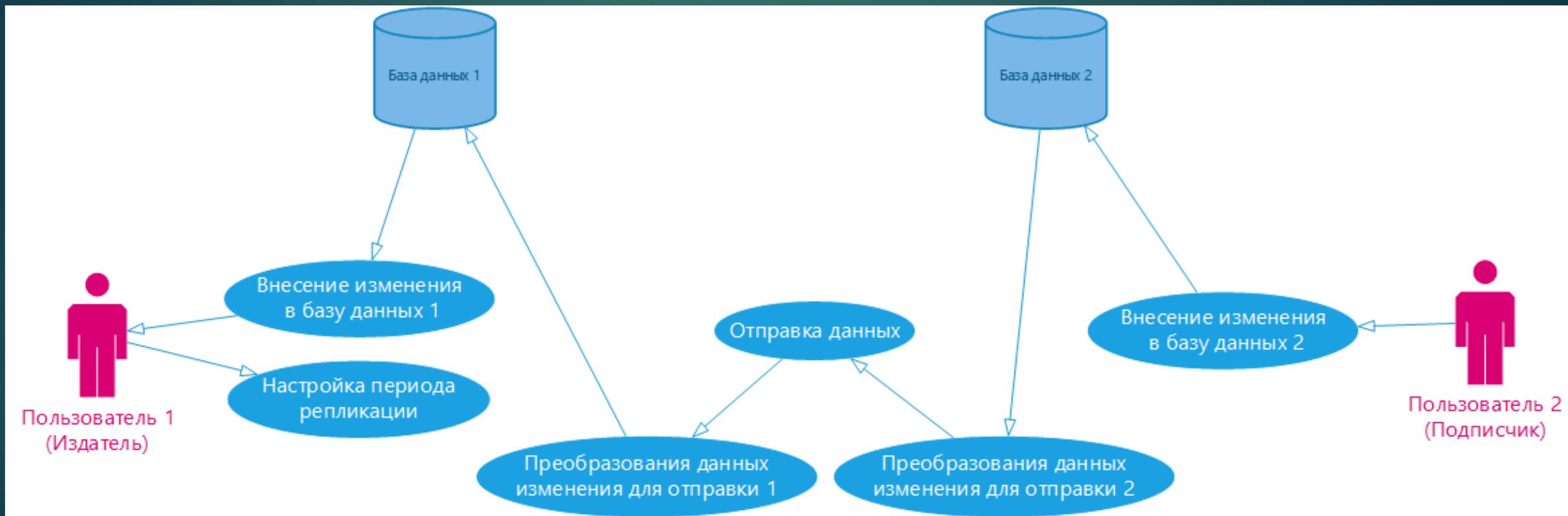
Сценарии использования

7



Сценарии использования

8



Отслеживание изменений

- ▶ 1) Отслеживание изменений с помощью триггеров
- ▶ 2) Отслеживание транзакций с помощью триггеров
- ▶ 3) Отслеживание изменений с помощью журнала

Файлы журнала и лога транзаций

10

```
I INTO users(name,pills) V ( 'John' , '3' );I
INTO users(name,pills) V ( 'Sam' , '10' );I
INTO users(name,pills) V ( 'Anna' , '4' );I
INTO users(name,pills) V ( 'Jack' , '14' );I
INTO users(name,pills) V ( 'Max' , '9' );I
INTO users(name,pills) V ( 'Kate' , '10' );U
users S name='Max' ,pills='9' W_id= '4' ;U
users S name='Anna' ,pills='3' W_id= '2' ;U
users S name='Kate' ,pills='9' W_id= '5' ;U
users S name='Max' ,pills='9' W_id= '4' ;|
```

Файл журнала транзакций

```
U users S name='Max' ,pills='9' W_id= '4' ;U
users S name='Anna' ,pills='3' W_id= '2' ;U
users S name='Kate' ,pills='9' W_id= '5' ;U
users S name='Max' ,pills='9' W_id= '4' ;I
INTO users(name,pills) V ( 'Kail' , '25' );I
INTO users(name,pills) V ( 'Ioan' , '3' );D
FROM users W_id= 6;|
```

Файл лога транзакций

Модуль репликации: SQLiteReplicationHelper

11

Метод	Функционал
<code>newInsert</code>	Формирует строку SQL запроса для вставки новых записей в базу данных, а также для добавления этой строки в файл логирования
<code>newDelete</code>	Формирует строку SQL запроса для удаления записей из базы данных, а также для добавления этой строки в файл логирования
<code>newUpdate</code>	Формирует строку SQL запроса для обновления записей в базе данных, а также для добавления этой строки в файл логирования
<code>queryBuilder</code>	Метод <code>queryBuilder</code> связывает значения выбора с аргументами
<code>getTypeOfObject</code>	Метод <code>getTypeOfObject</code> возвращает тип данных объекта

Модуль репликации: SingleLine

12

Метод	Функционал
saveToFile	Сохраняет строку SQL запроса в файл журнала и лога транзакций
readFromLogForSend	Возвращает из файла лога транзакций строку со всеми данными и преобразует их для дальнейшей отправки
convertSMS	Принимает от обработчика входящих SMS сообщений строку и преобразует ее для обновления базы данных
readAllFromJournal	Возвращает из файла журнала транзакций строку со всеми запросами
clearLog	Очищает файл лога транзакций
clearJournal	Очищает файл журнала транзакций

Метод внесения строки в таблицу (INSERT)

```
database.insert(DBHelper.TABLE_NAME, null, contentValues);
```

```
database.execSQL(replicationHelper.newInsert(DBHelper.TABLE_NAME  
    , null, contentValues, this));
```

Метод изменения строки в таблице (UPDATE)

14

```
database.update(DBHelper.TABLE_NAME, contentValues,  
    DBHelper.KEY_ID + "= ?", new String[]{id});
```

```
database.execSQL(replicationHelper.newUpdate(DBHelper.TABLE_NAME,  
    contentValues, DBHelper.KEY_ID + "= ?", new String[]{id}, this));
```

Метод удаления строки из таблицы (DELETE)

```
database.delete(DBHelper.TABLE_NAME, DBHelper.KEY_ID + "=",  
               new String[]{id});
```

```
database.execSQL(replicationHelper.newDelete(DBHelper.TABLE_NAME,  
                                             DBHelper.KEY_ID + "=", new String[]{id}, this));
```

Демонстрационное приложение

16

Android Emulator - 5.1_WVGA_API_28:5554

6:33

Номер телефона

ОТПРАВИТЬ SMS

ПОДТВЕРДИТЬ НОМЕР

ОЧИСТИТЬ ФАЙЛ

Имя

Количество таблеток

ДОБАВИТЬ ПРОЧИТАТЬ УДАЛИТЬ ВСЕ ДАННЫЕ

ИД

ОБНОВИТЬ УДАЛИТЬ ПО ИД

СОХРАНИТЬ ДАТУ

Выкл

< May 2020 >

S	M	T	W	T	F	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

Android Emulator - 5.1_WVGA_API_28:5554

6:34

Все данные таблицы:

id: 1 name: Sam pills: 1
id: 2 name: Anna pills: 4
id: 3 name: Jack pills: 9
id: 4 name: Max pills: 12
id: 5 name: Kate pills: 10

Тестирование

- ▶ Интеграционное тестирование на реальных и эмулированных мобильных телефонах.
- ▶ Системное тестирование, а именно альфа-тестирование реализованного функционала.

Перспективы развития

- ▶ Вместо использования службы коротких сообщений использовать популярные мессенджеры, такие как: Viber, Whatsapp и др.