Трудности разработки файлового менеджера для iOS

Веденеев Игорь, AGIMA

Mobius, November 2020

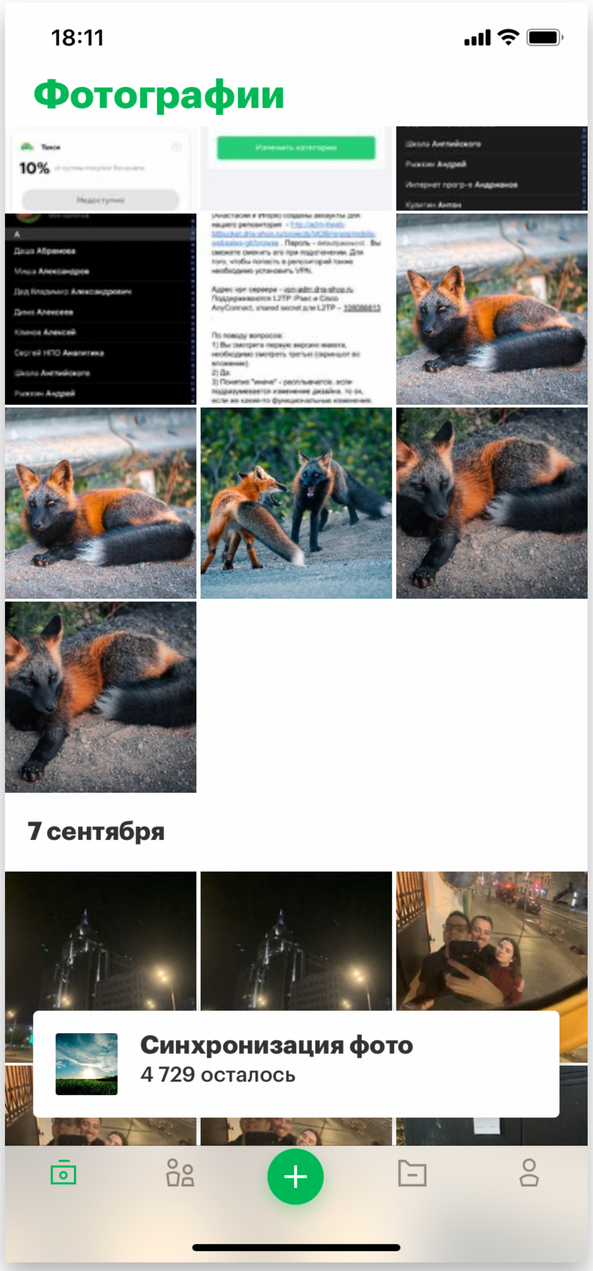
AGIMA

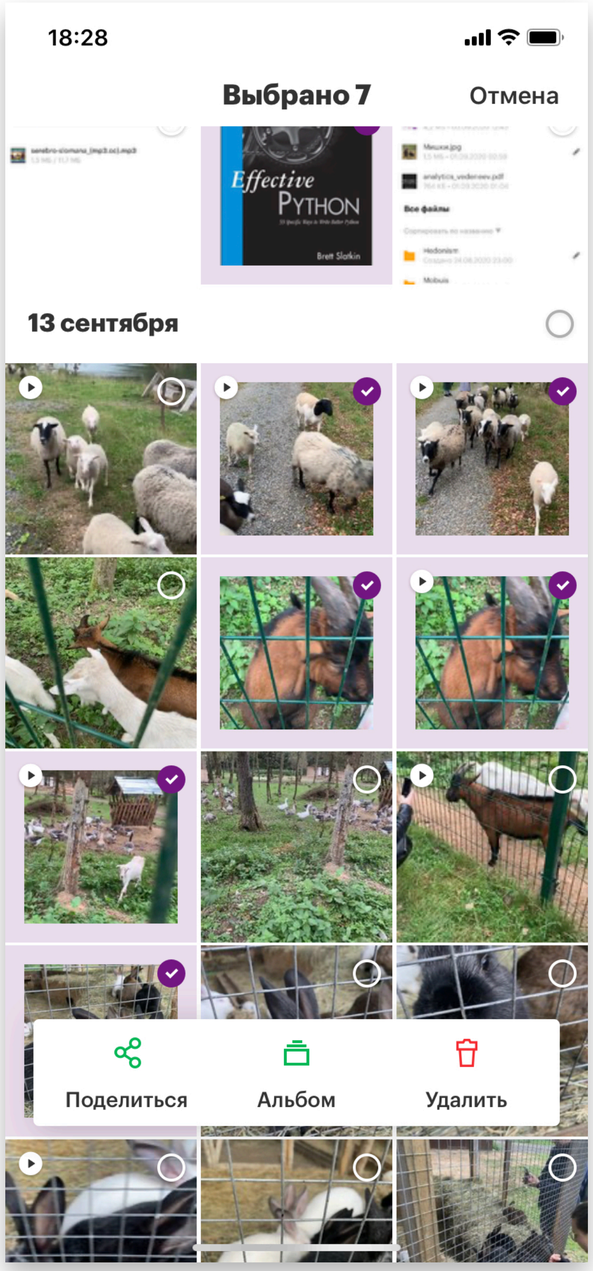
Файловый менеджер

Это

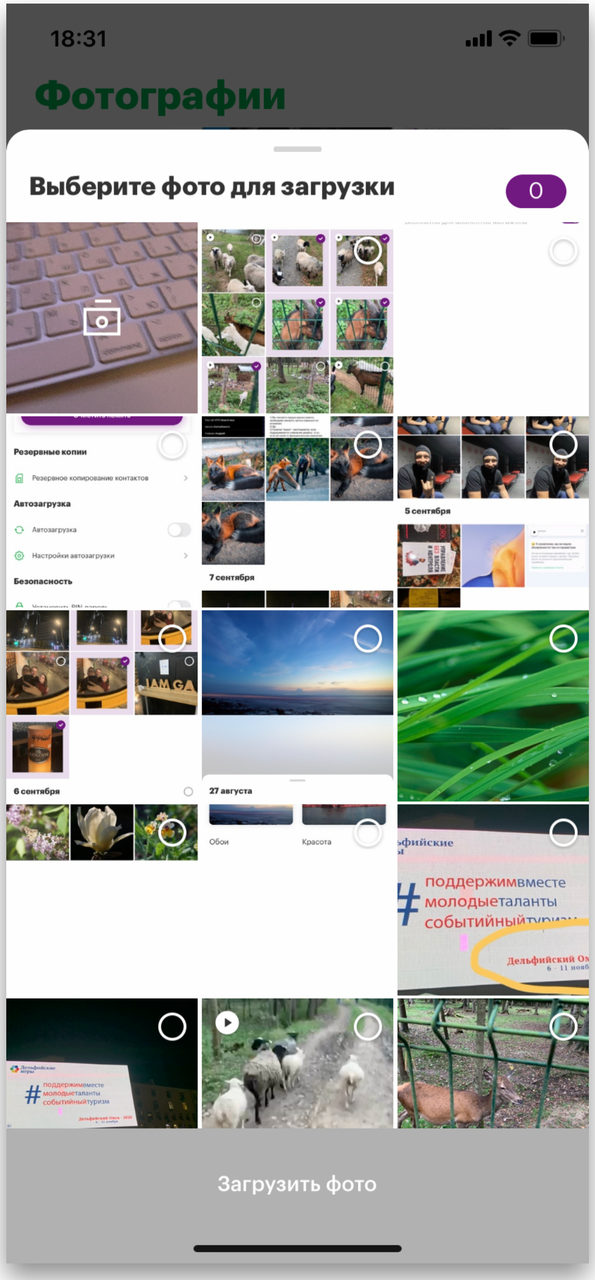
- Загрузка файлов на сервер
- Просмотр файлов разных форматов
- Просмотр/изменение иерархии папок и файлов
- Скачивание файлов
- io
- Просмотр и взаимодействия с медиа*

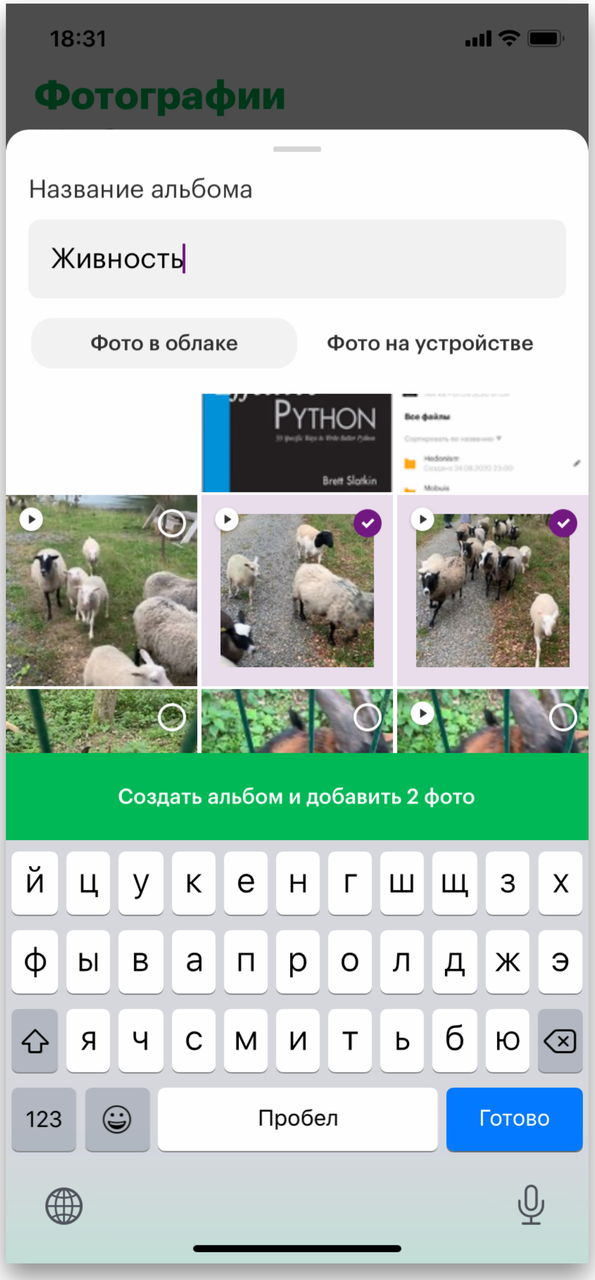
Лента фото


18:11
Фотографии
7 сентября
Синхронизация фото
4 729 осталось

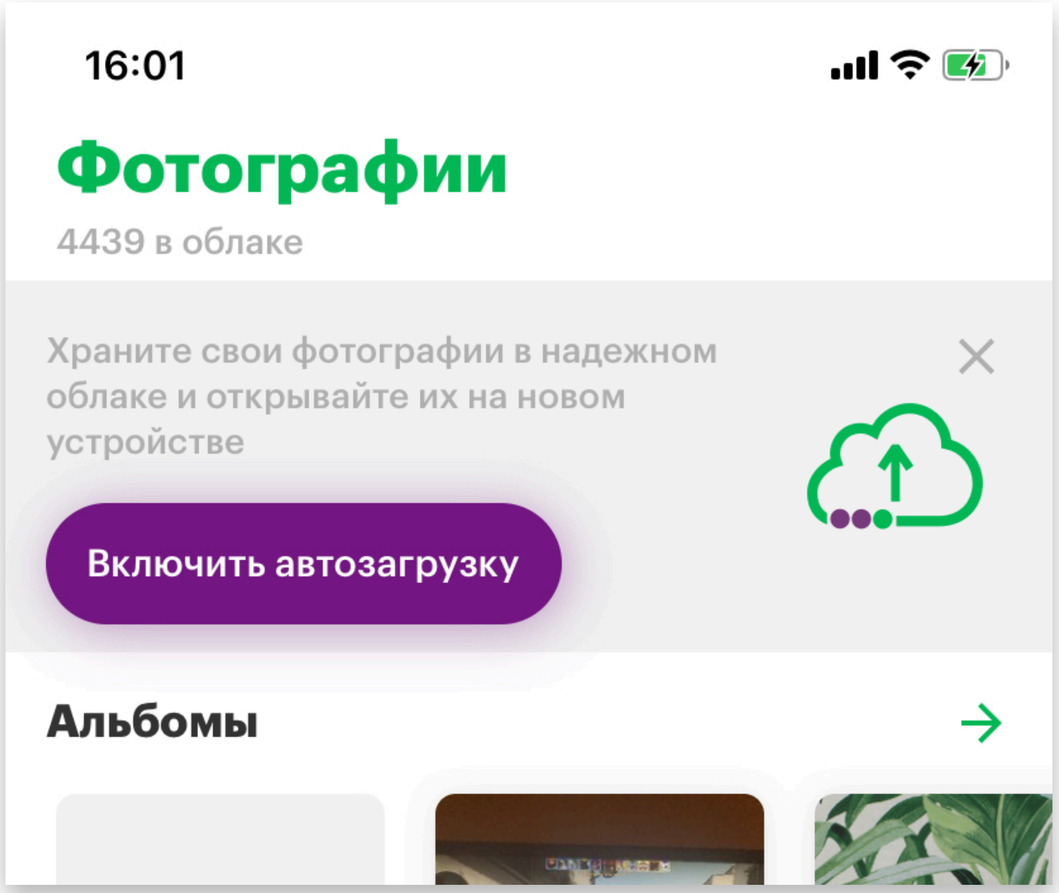

18:28
Выбрано 7
Отмена
13 сентября
Поделиться
Альбом
Удалить

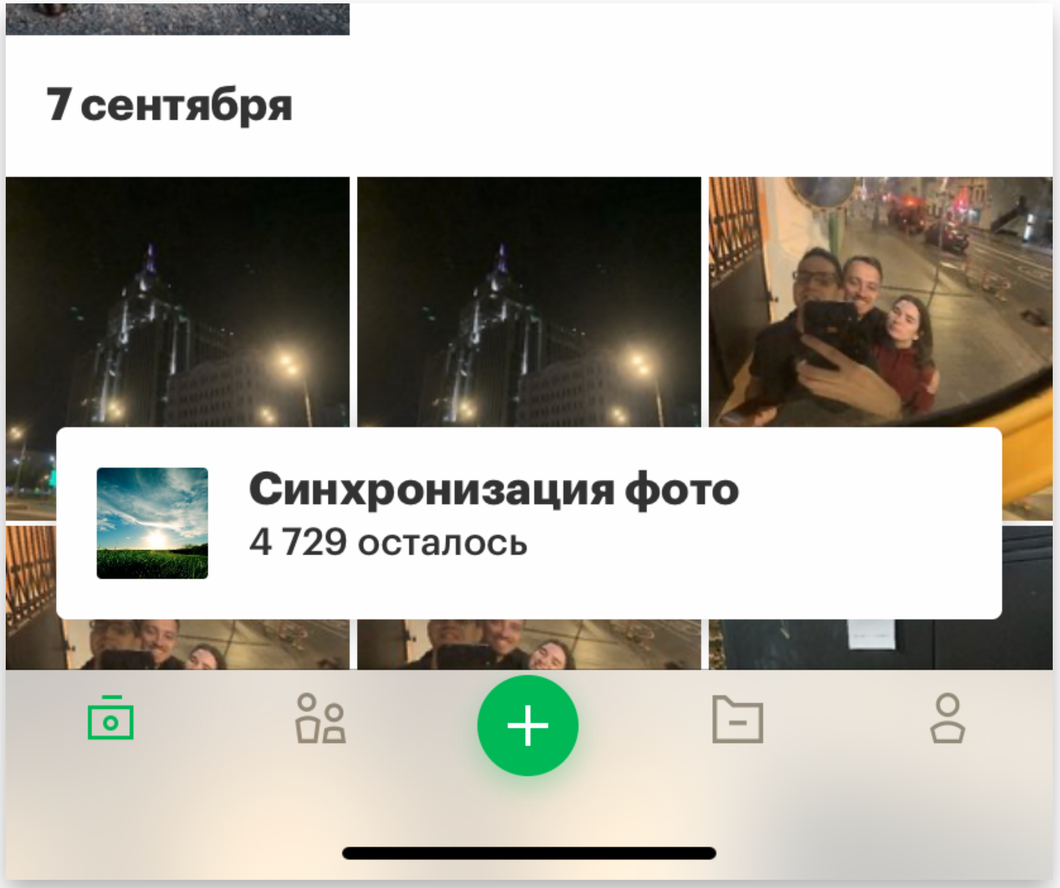
Загрузка фото



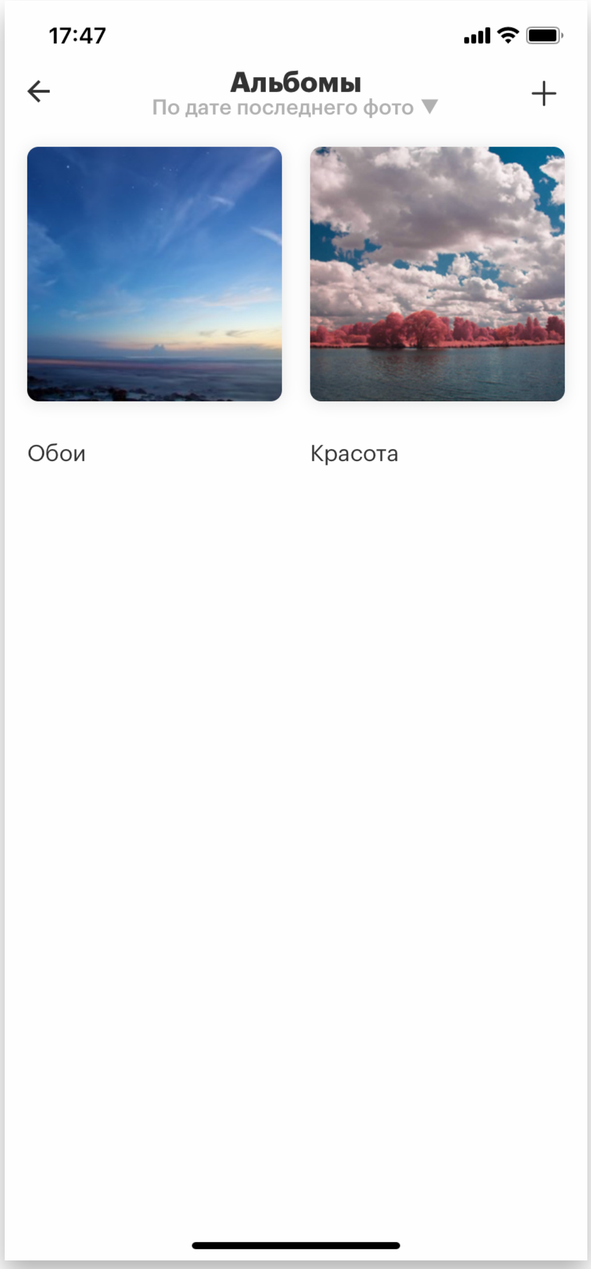


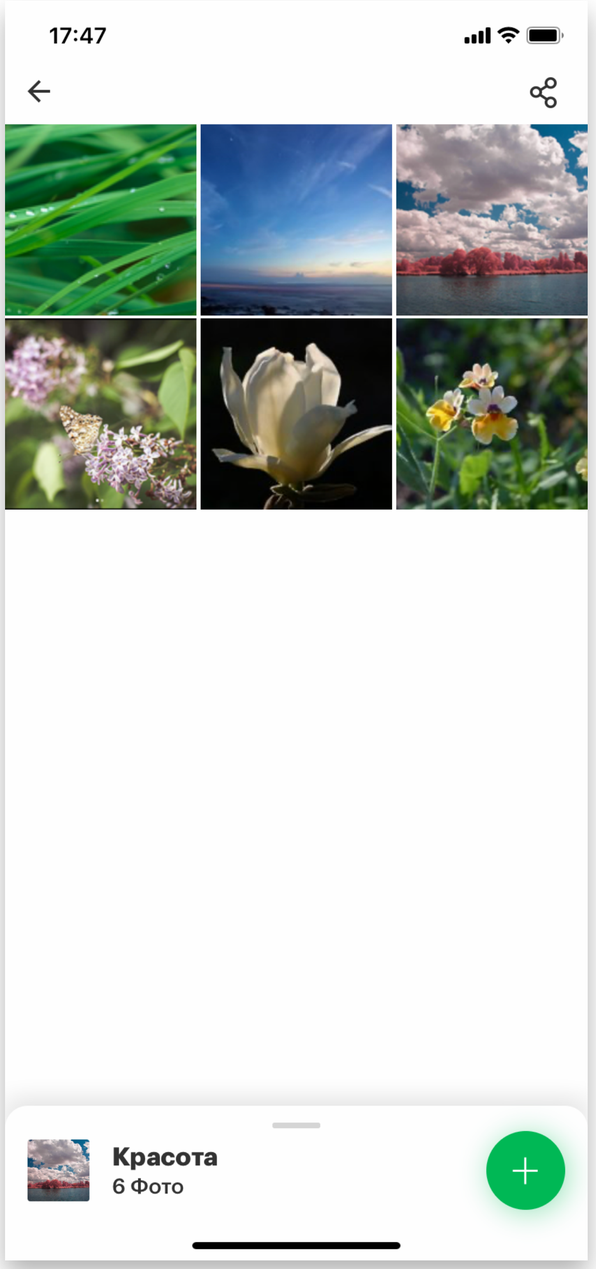
Автозагрузка фото



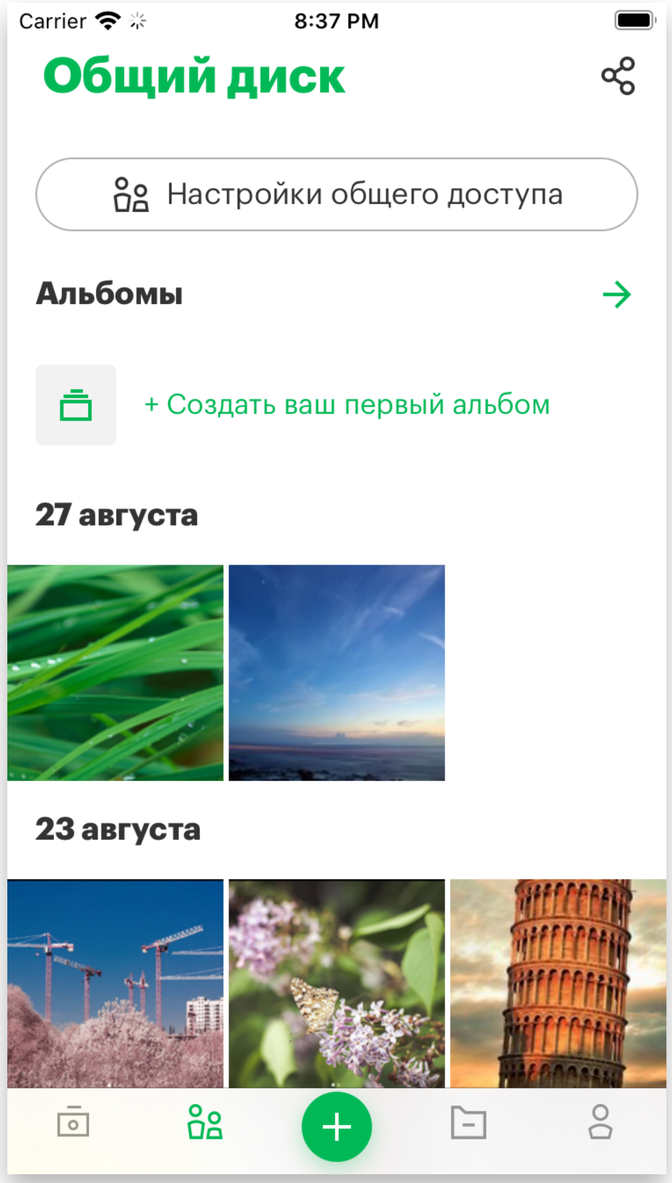


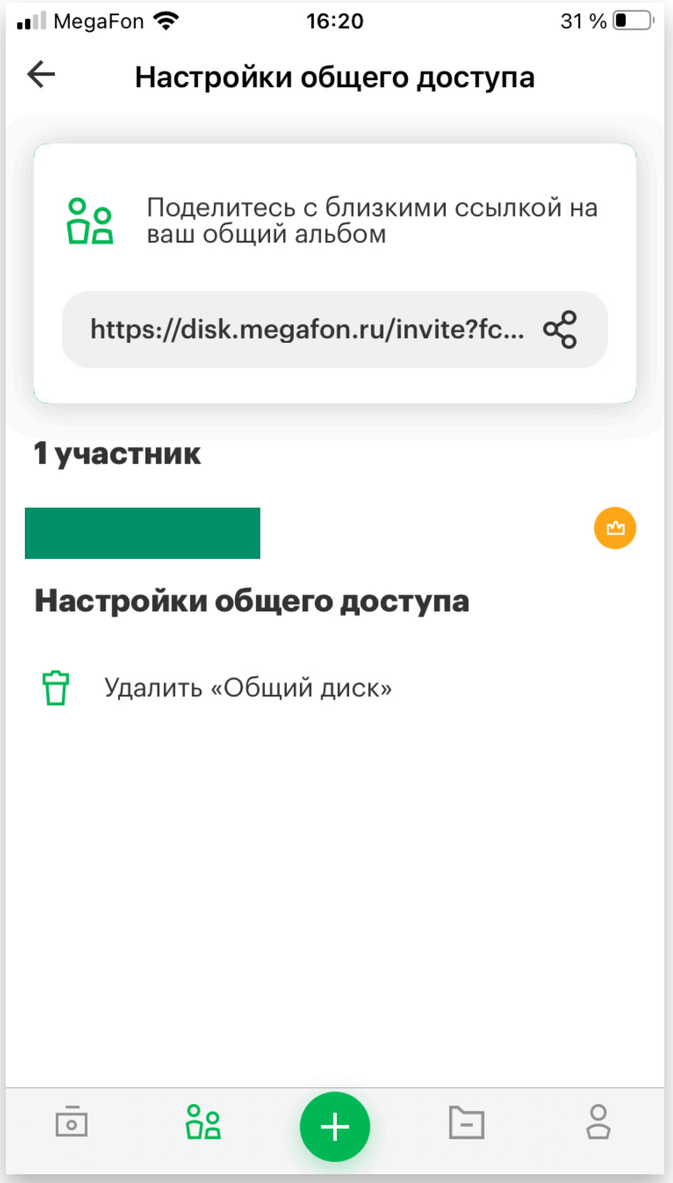
Альбомы


17:47
Альбомы
По дате последнего фото
Обои
Красота

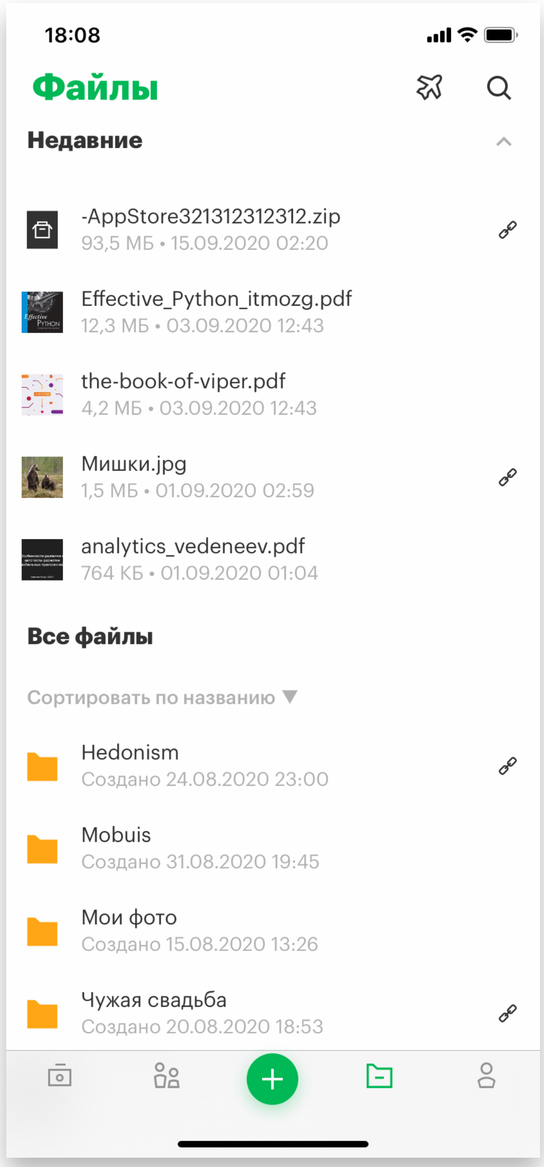

17:47
Красота
6 Фото

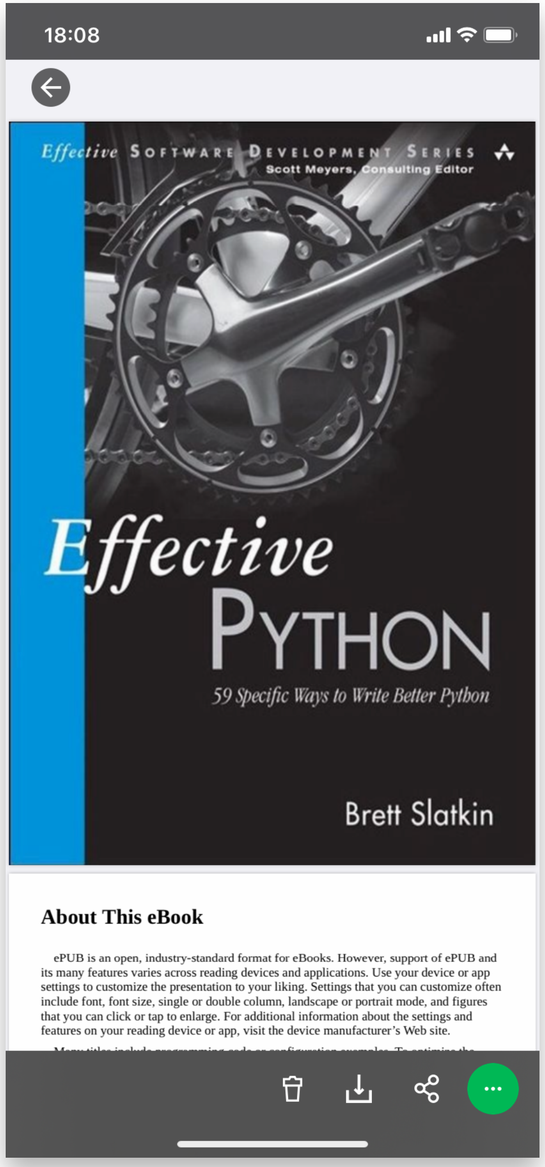
Общий диск

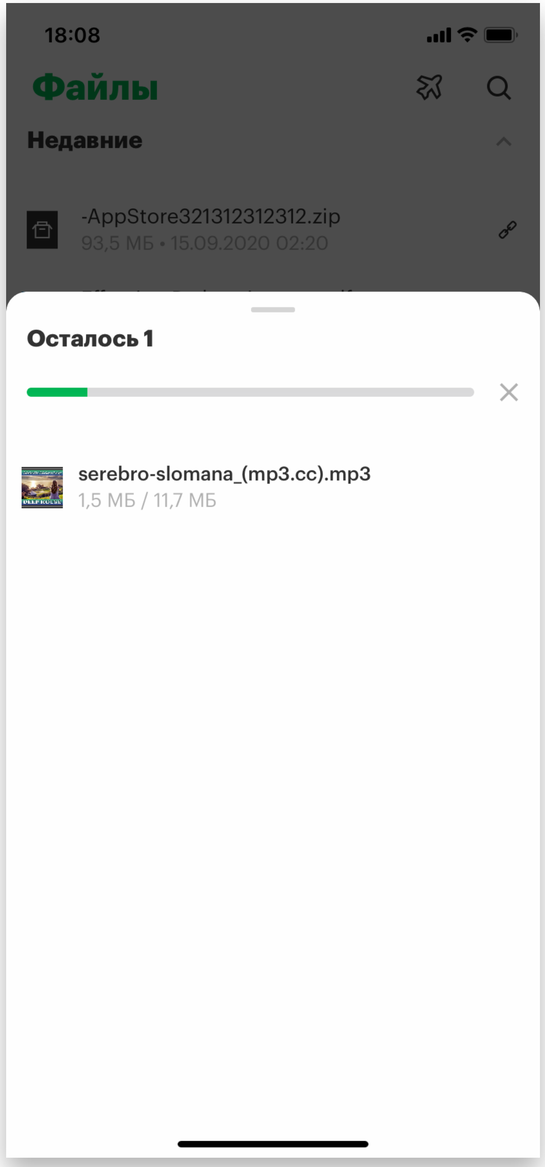

Carrier
8:37 PM
Общий диск
Настройки общего доступа
Альбомы
+ Создать ваш первый альбом
27 августа
23 августа


MegaFon
16:20
31 %
Настройки общего доступа
Поделитесь с близкими ссылкой на ваш общий альбом
https://disk.megafon.ru/invite?fc...
1 участник
Настройки общего доступа
Удалить «Общий диск»

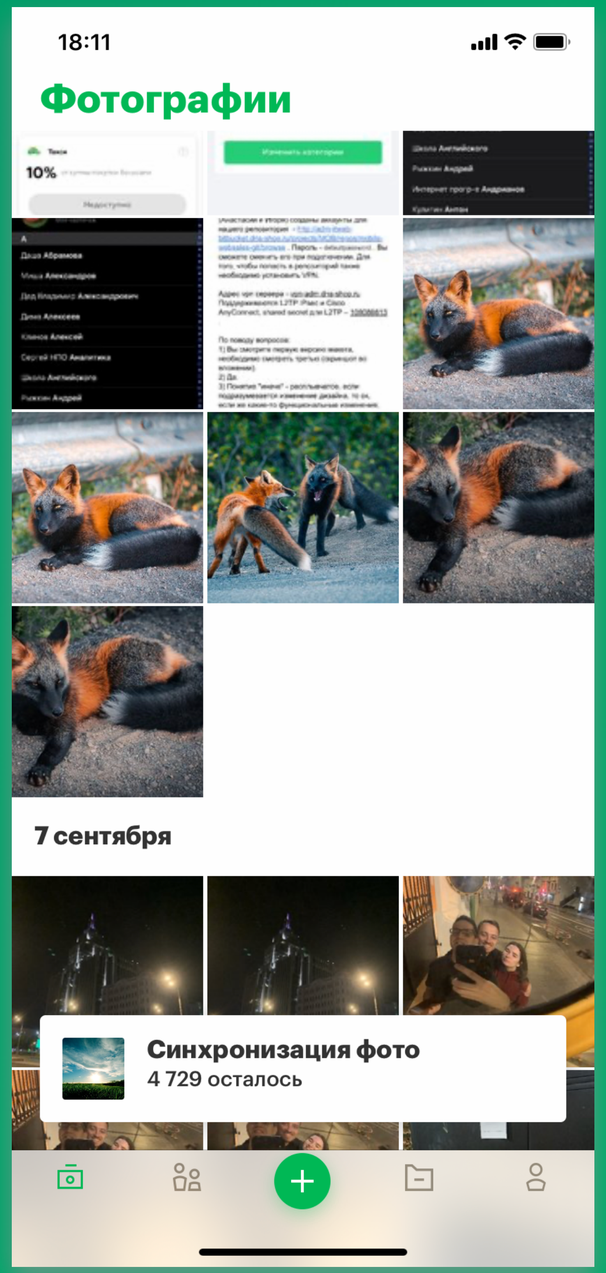
Файлы


18:08
Файлы
Недавние
-AppStore321312312312.zip
93,5 МБ • 15.09.2020 02:20
Effective_Python_itmozg.pdf
12,3 МБ • 03.09.2020 12:43
the-book-of-viper.pdf
4,2 МБ • 03.09.2020 12:43
Мишки.jpg
1,5 МБ • 01.09.2020 02:59
analytics_vedeneev.pdf
764 КБ • 01.09.2020 01:04
Все файлы
Сортировать по названию
Hedonism
Создано 24.08.2020 23:00
Mobuis
Создано 31.08.2020 19:45
Мои фото
Создано 15.08.2020 13:26
Чужая свадьба
Создано 20.08.2020 18:53


18:08
Effective Software Development Series
Scott Meyers, Consulting Editor
Effective
PYTHON
59 Specific Ways to Write Better Python
Brett Slatkin
About This eBook
ePUB is an open, industry-standard format for eBooks. However, support of ePUB and its many features varies across reading devices and applications. Use your device or app settings to customize the presentation to your liking. Settings that you can customize often include font, font size, single or double column, landscape or portrait mode, and figures that you can click or tap to enlarge. For additional information about the settings and features on your reading device or app, visit the device manufacturer's Web site.


18:08
Файлы
Недавние
-AppStore321312312312.zip
93,5 МБ • 15.09.2020 02:20
Осталось 1
serebro-slomana_(mp3.cc).mp3
1,5 МБ / 11,7 МБ

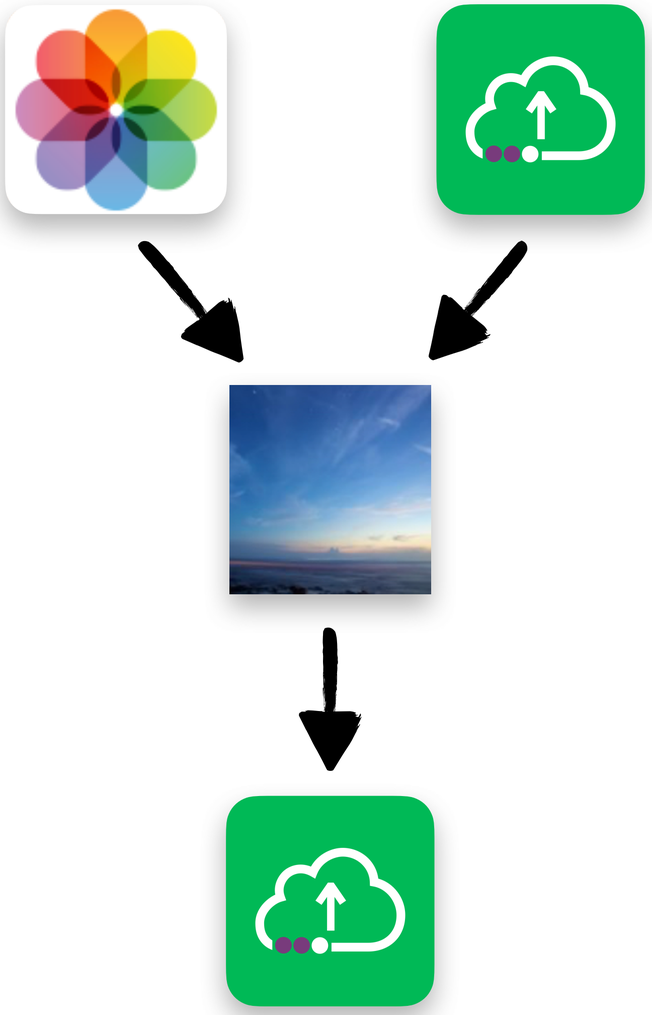
Лента фото


18:11
Фотографии
7 сентября
Синхронизация фото
4 729 осталось

Особенности

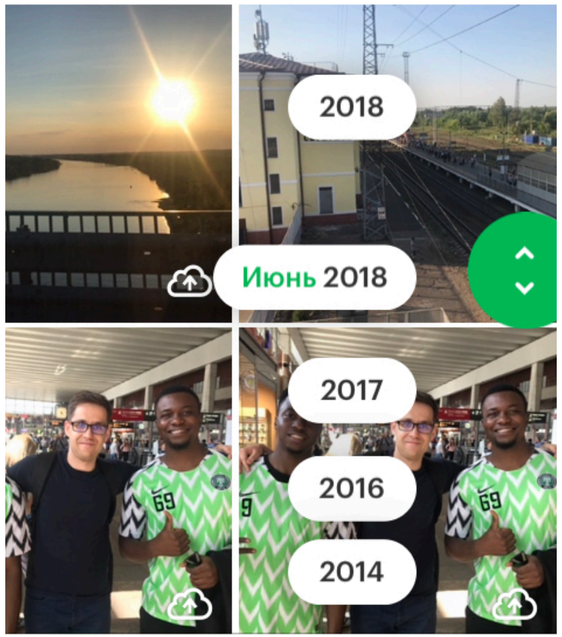
Лента фото

Особенности



Особенности

Особенности



Проблемы

- Много источников данных (облако, галлерея, альбомы…)
- **Лента долго группируется по дням**
- `NSInternalInconsistencyException`

Источники данных

Облако

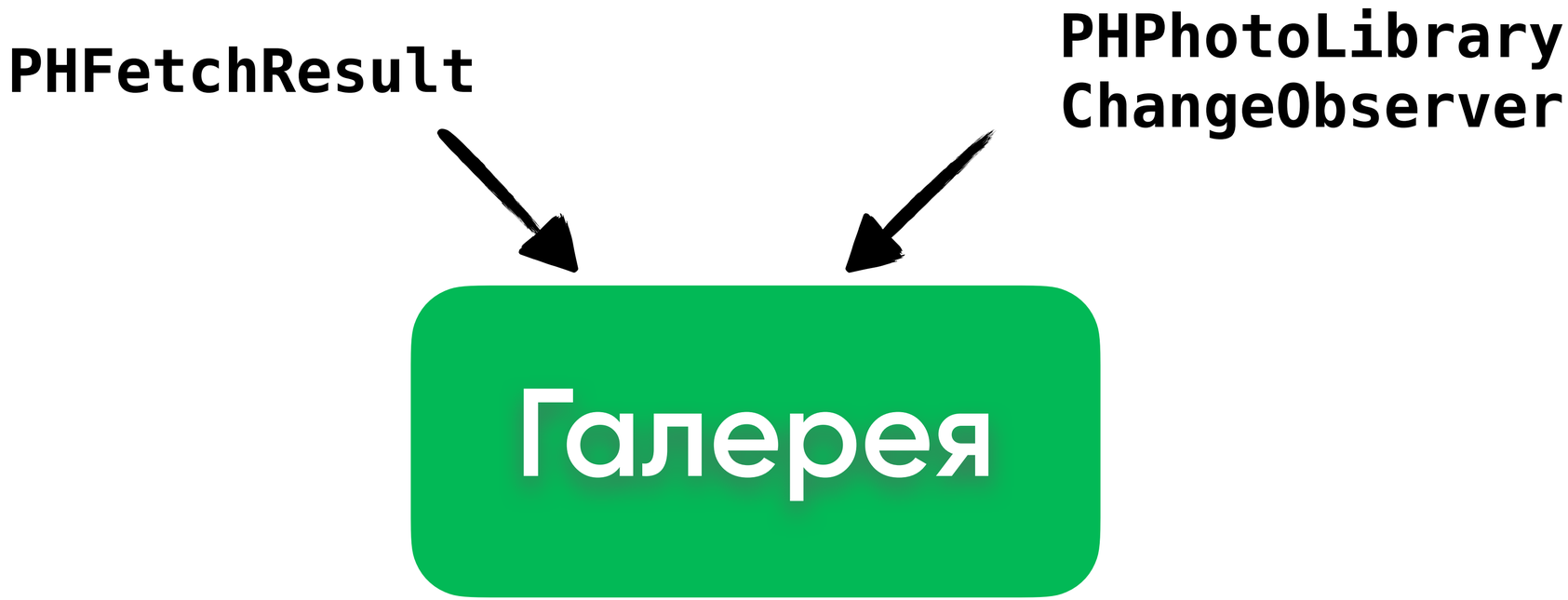
Автозагрузка

Альбомы

Галерея

Источники данных

Галерея

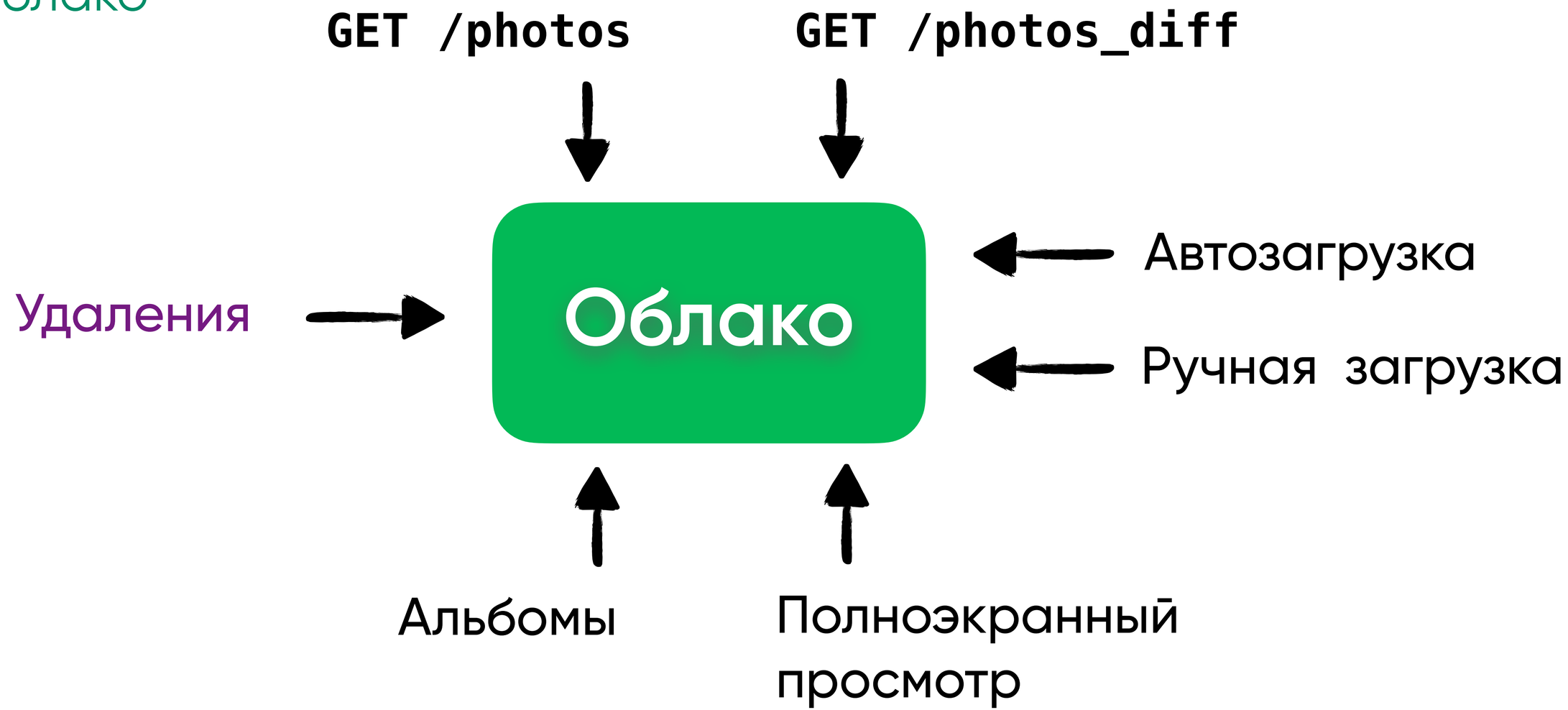


Источники данных

Облако

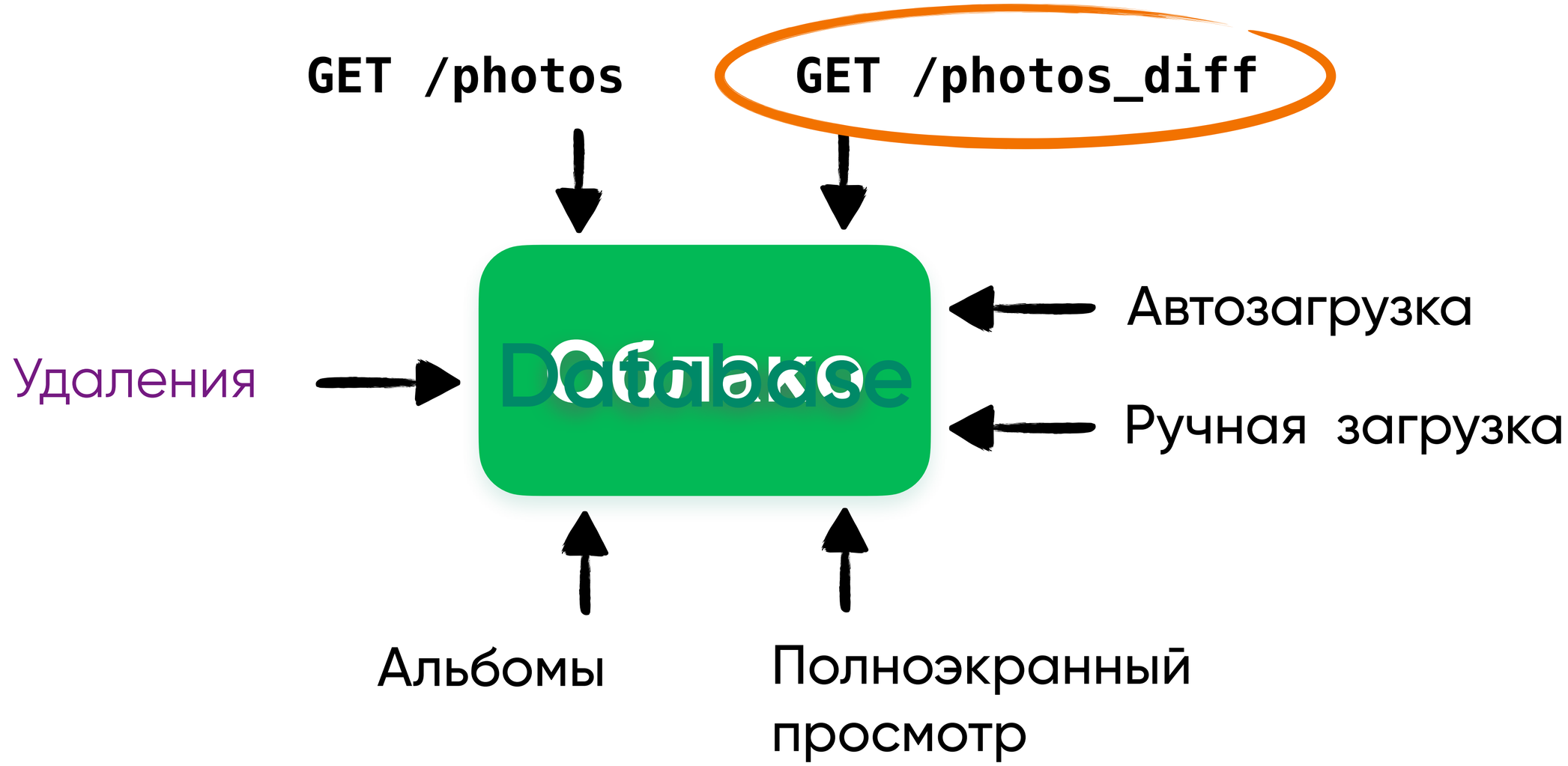
Источники данных

Облако



Источники данных

Источники данных



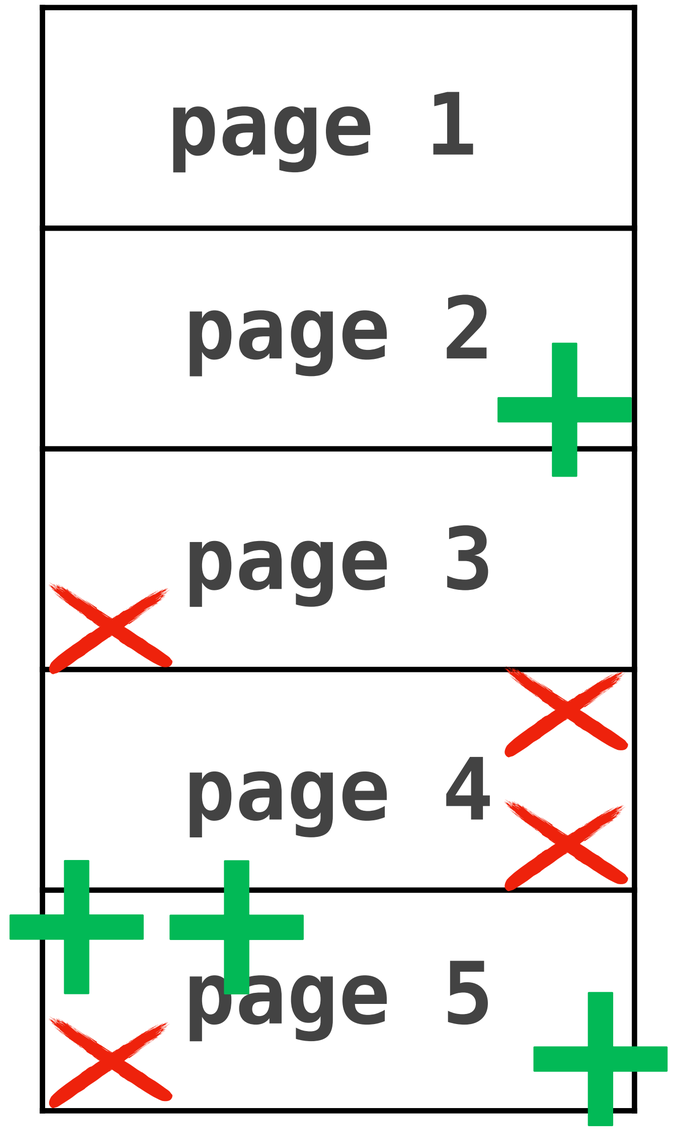
API

/photos

Список фото

/photos_diff

Список операций
с фото



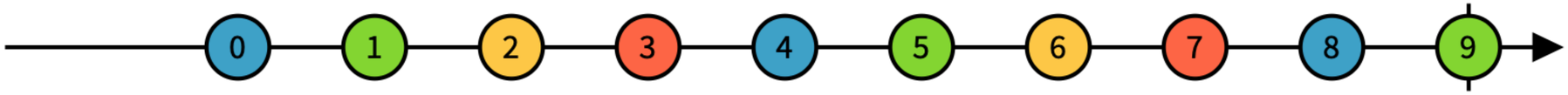
API

Запуск 1	`/photos`
Запуск 2	`/photos_diff?from_date=date_1`
Запуск 3	`/photos_diff?from_date=date_2`
Запуск N	`/photos_diff?from_date=date_n`

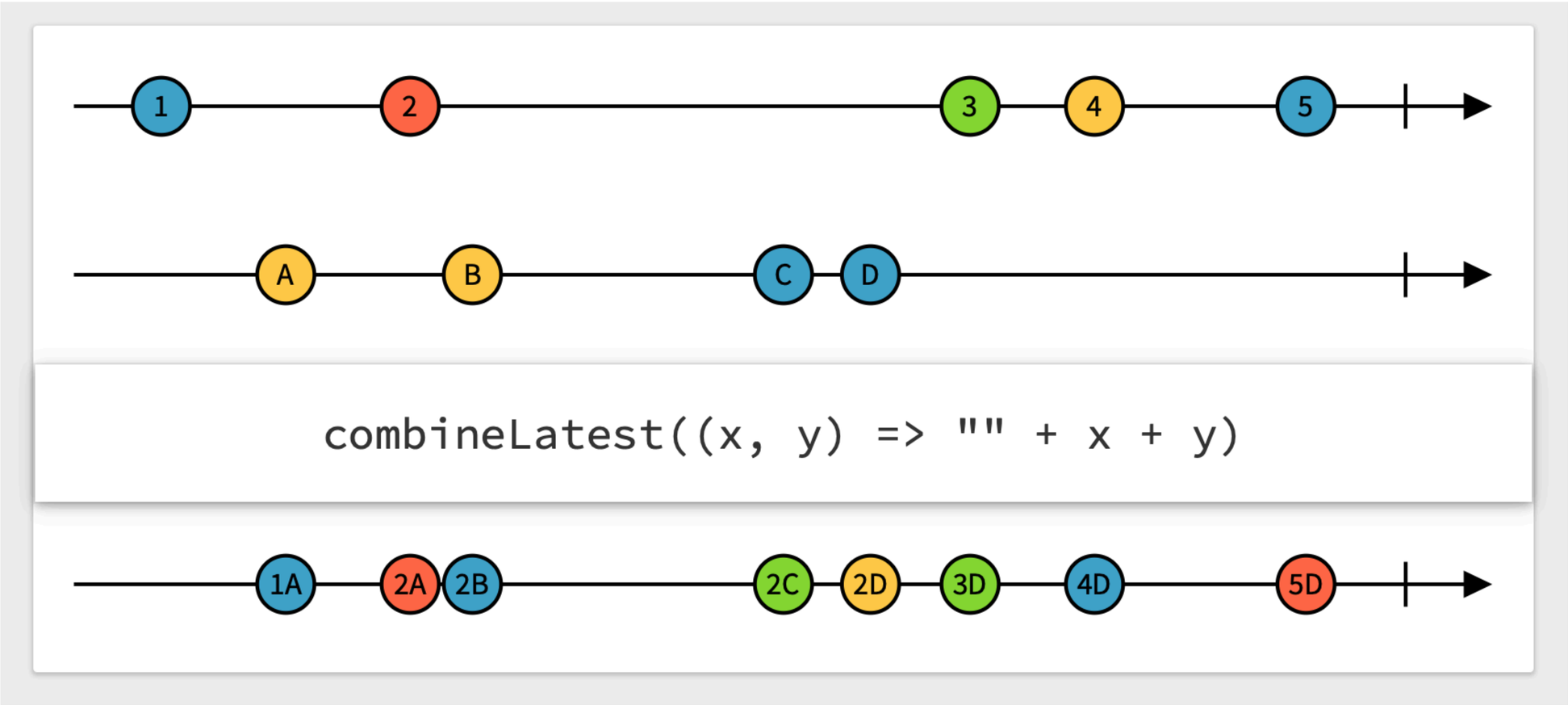

ReactiveSwift

ReactiveSwift

```
Signal<T, Error>
```



ReactiveSwift



Много источников данных

Объединяем в один

```
Signal.combineLatest(
    photoChanges,
    albumChanges,
    galleryChanges,
    turnOnAutosync.take(first: 1)
)
```

ReactiveSwift

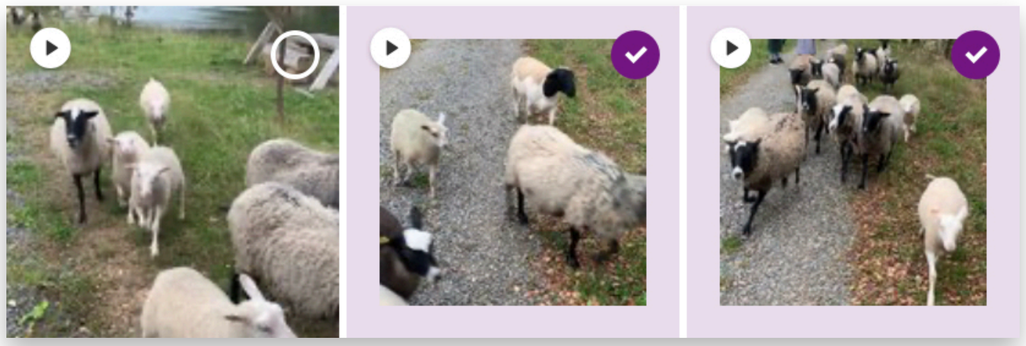
```
class Property {
    let value: Value { get }
    let signal: Signal<Value, Never>
}


class MutableProperty {
    let value: Value { get set }
    let signal: Signal<Value, Never>
}
```

Долго группируется по дням

Ошибки

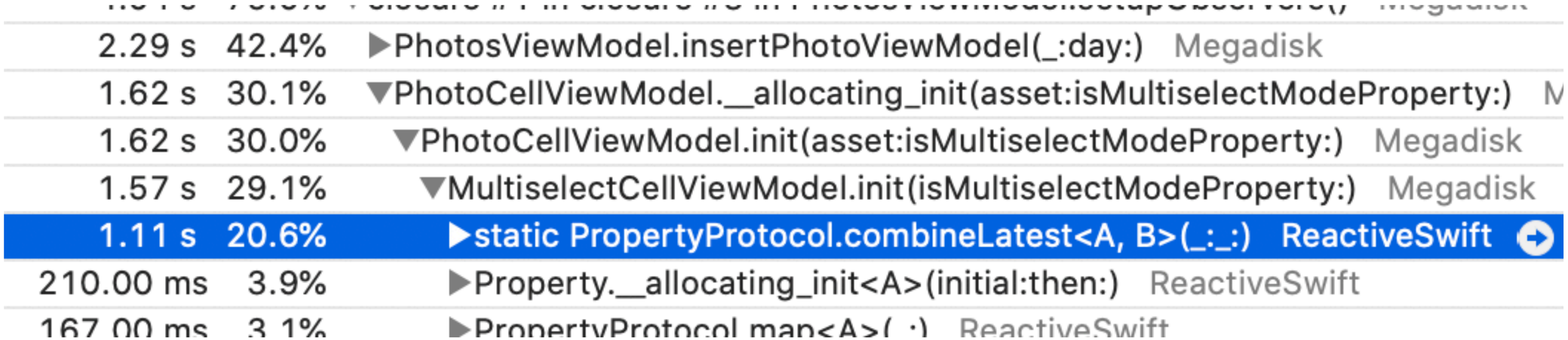
```
class PhotoCellViewModel {
    let isMultiselectMode: Property<Bool>
    let isMultiselected: Property<Bool>
}
```

- Тратится время на создание для `Property` каждой ячейки



Долго группируется по дням

Ошибки



Долго группируется по дням

Решение

```
enum MultiselectState {
    case `default`
    case multiselected(isSelected: Bool)
}

var onStateChange: ((MultiselectState) -> Void)?
```

Долго группируется по дням

```
class PhotoSection {
    var photos: [PhotoCellViewModel]
    let date: Date
}
```

Долго группируется по дням

Решение

```
if есть_фотка_в_секции {
    // удалить старую модель
    // вставить новую
}
```

```
if есть_фотка_в_секции {
    // изменить стейт у модели
    // оповестить об изменении
}
```

Долго группируется по дням

Решение

```
class PhotoCellViewModel {
    var onSyncChange: ((Bool) -> Void)?
}
```

Долго группируется по дням

Обработка событий

```
class PhotosSection {
    var photos = [PhotoCellViewModel]()
    let date: Date

    let didSelect: Signal<T, Never>
    let didMultiselect: Signal<T, Never>
    let didMultideselect: Signal<T, Never>
}
```

- Секций много
- Требуется время на создание сигналов и подписку
- Секции добавляются / удаляются

Долго группируется по дням

Обработка событий

```
class PhotosSection {
    let didSelect: Signal<T, Never>
    let didMultiselect: Signal<T, Never>
    let didMultideselect: Signal<T, Never>
}
```

```
class ModernPhotoSection {
    weak var delegate: ModernPhotoSectionDelegate?
}
```

Долго группируется по дням

Обработка событий

```
class ModernPhotoSection {
    weak var delegate: ModernPhotoSectionDelegate?
}
```

- Не важно, сколько будет секций и будет ли меняться список
- Проставить делегат быстрее чем создать сигналы и подписаться

Rx 💔?

Скорость формирования ленты фото

*8000 фото в облаке и 5000 фото в галерее, iPhone SE

Было	Оптимизизация	Убрали галерею
~4с.	~1с.	~0.5с.

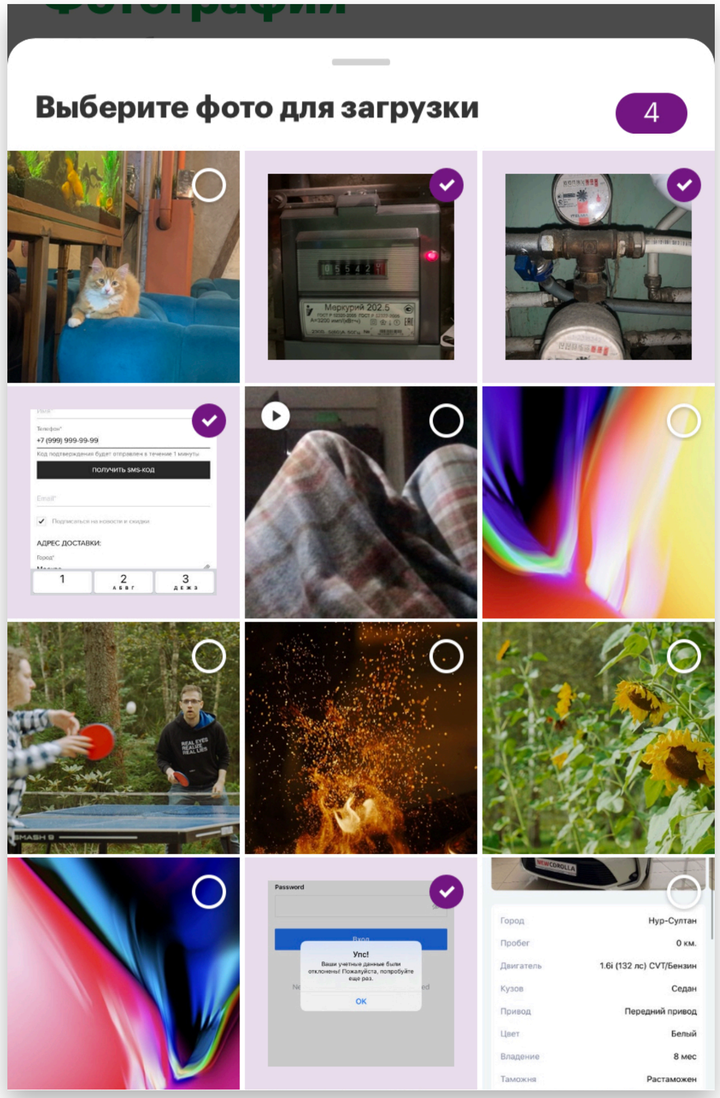
Светлое настоящее

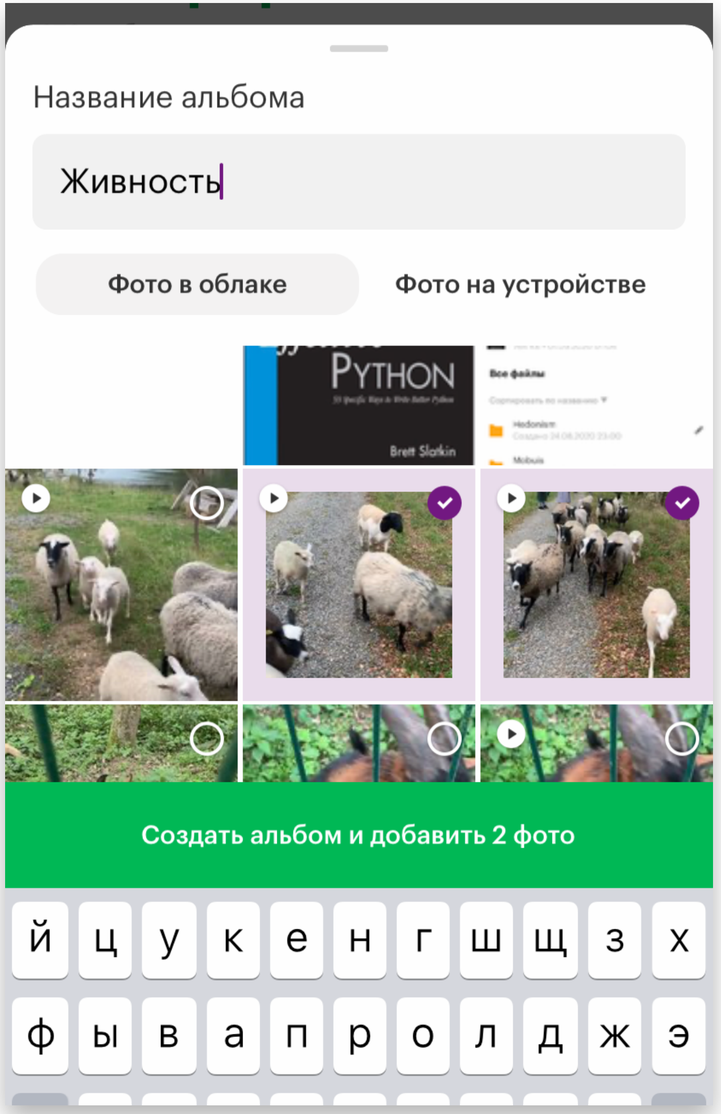
Убрали фото из галлереи

```
class PhotoDay {
    var date: Date
    var photos: [Photo]
}
```

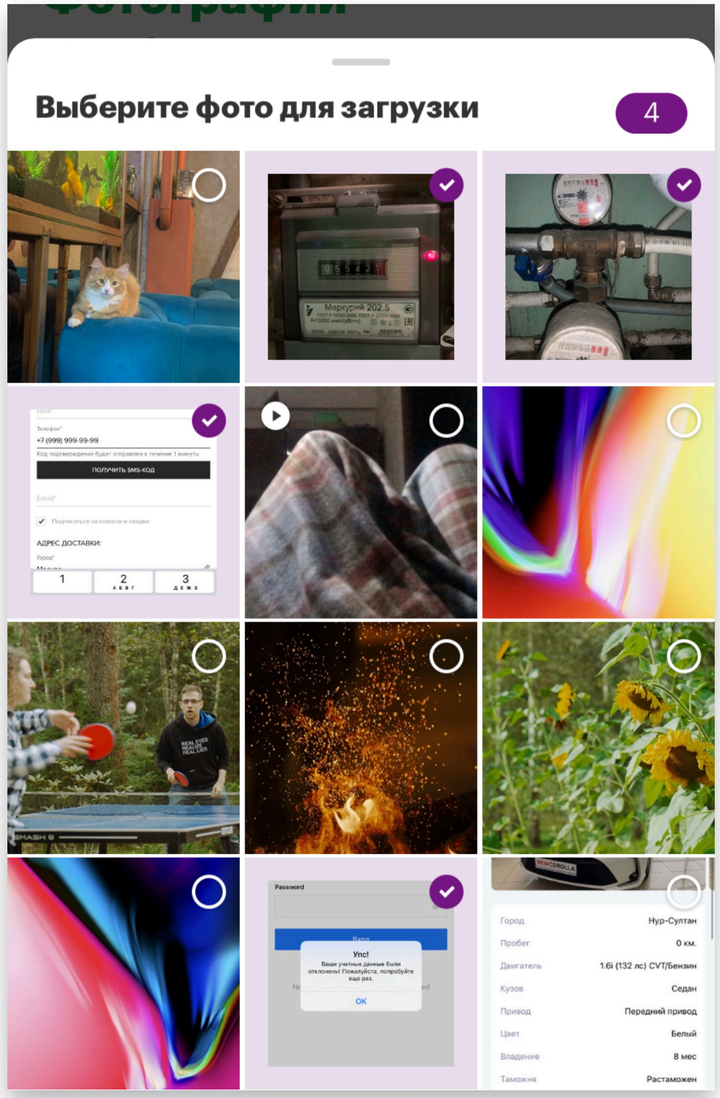
Фото и видео из галереи вашего телефона больше не будут отображаться в МегаДиске

Еще ленты


Выберите фото для загрузки
4


Название альбома
Живность
Фото в облаке
Фото на устройстве
Создать альбом и добавить 2 фото

Еще ленты



PHFetchResult

Скорость формирования ленты фото из галереи

*15 000 фото

Было	Стало
~2сек	~0.2сек

x10

Выводы

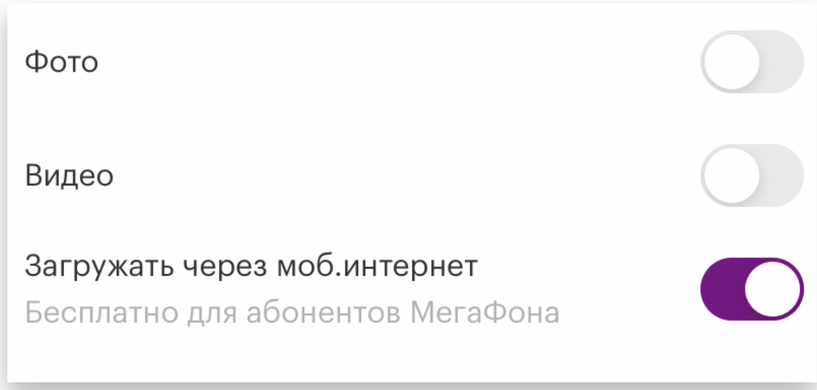
Лента фото

- Edge-кейсы будут
- “Одну и ту же” задачу можно решать по-разному
- Потребляемая память

Автозагрузка фото

Задачи

- Загрузить все фото из галереи в “фоновом” режиме
- Автозагрузка в бекграунде
- Не загружать дубли*
- Не загружать удалённые фото
- Обрабатывать изменения из галлереи
- Синхронизироваться с “ручной” загрузкой
- Учитывать настройки пользователя и др. условия
- *Загружать фото в несколько потоков*



Автозагрузка

Триггеры

```
let toggleSettings: Signal<Bool, Never>
let acceptedMeidaTypes: Signal<MediaType, Never>
let toggleLowPowerMode: Signal<Bool, Never>
let toggleCelluarUploadSettings: Signal<Bool, Never>
let networkConditions: Signal<ConnectionType, Never>
let silentPushTrigger: Signal<Void, Never>
let backgroundURLSessionEvents: Signal<Void, Never>
let didFinishFetchPhotosFromCloud: Signal<Void, Never>
let diskIsFull: Signal<Bool, Never>
```

Автозагрузка

Триггеры

```
Signal.combineLatest(...)
  .map { (s, med, lowp, clr, cond, _) -> (Bool, MediaType) in
     let conditionsAreOk = clr || cond == .wifi
     let isOn = s && !lowp && conditionsAreOk
     return (isOn, med)
  }
  .merge(with: diskIsFull)
  .skipRepeats()
```

Очередь загрузки

Как было

- Запускаем операции последовательно
- Каждая операция это `SignalProducer`
- Добавлять/удалять неприятно

```
SignalProducer.concat(uploadProducers)
    .startWithResult { (result) in
        // handle results
    }
```

Очередь загрузки

maxConcurrentCount на ReactiveSwift

```
SignalProducer(producers)
    .flatten(.concurrent(limit: 2))
    .startWithResult { (result) in
       // handle results
    }
```

```
1 start
2 start
1 finish
3 start
2 finish
4 start
3 finish
5 start
4 finish
6 start
5 finish
6 finish
```

Очередь загрузки

Очередь...🧐

…`OperationQueue`!

- Операции не зависят друг от друга
- `maxConcurrentOperationsCount`
- Нет проблем с добавлением / удалением операций
- Есть отмена операций
- Серьезный рефакторинг

Не загружать дубли и лишние фотки

- Синхронизация ручной и автозагрузки
- Не можем расчитывать на `PHAsset.localIdenttifier`

☑ Вычисление hash на клиенте и сервере

Вычисление hash на устройстве

`localIdentifier` на то и `local`...

`PHImageManager.requestImage` выдает разные hash

`requestImageDataAndOrientation`

PHAsset fetch. It's never easy

PHAssetResourceType

```
public enum PHAssetResourceType : Int {
    case photo = 1
    case video = 2
    case adjustmentData = 7
}
```

PHAsset fetch. It's never easy

PHAssetResourceType

```
(lldb) po PHAssetResource.assetResources(for: phAsset)
▿ 3 elements
  - 0 : <PHAssetResource: 0x600000ecbe70> {
    type: adjustment
    uti: com.apple.property-list
    filename: Adjustments.plist
    fileURL: file:///.../Adjustments.plist
}
```

Разный порядок элементов на iOS12 и iOS13

Вычисление hash на устройстве

Hash для видео

```
let bufferSize = 1024 * 1024
guard let file = try? FileHandle(forReadingFrom: self) else { return nil }

defer {
    file.closeFile()
}

var context = CC_SHA1_CTX()
CC_SHA1_Init(&context)
while autoreleasepool(invoking: {
    let data = file.readData(ofLength: bufferSize)
    if data.count > 0 {
        data.withUnsafeBytes {
            _ = CC_SHA1_Update(&context, $0.baseAddress, numericCast(data.count))
        }
        return true
    } else {
        return false
    }
}) {}

var digest = [UInt8](repeating: 0, count: Int(CC_SHA1_DIGEST_LENGTH))
_ = CC_SHA1_Final(&digest, &context)

return Data(digest)
```


Вычисление hash на устройстве

Доступ к файлу

```
let bufferSize = 1024 * 1024
guard let file = try? FileHandle(forReadingFrom: self)
else { return nil }

defer {
    file.closeFile()
}
```

Вычисление hash на устройстве

Initialize

```
var ctx = CC_SHA1_CTX()
CC_SHA1_Init(&ctx)
```

Вычисление hash на устройстве

Calculate hash

```
while autoreleasepool(invoking: {
    let data = file.readData(ofLength: bufferSize)
    if data.count > 0 {
        data.withUnsafeBytes {
            CC_SHA1_Update(&ctx, $0.baseAddress, numericCast(data.count))
        }
        return true
    } else {
        return false
    }
}) {}
```

Вычисление hash на устройстве

Finalize

```
var digest = [UInt8](repeating: 0, count: Int(CC_SHA1_DIGEST_LENGTH))
_ = CC_SHA1_Final(&digest, &context)
```

Автозагрузка

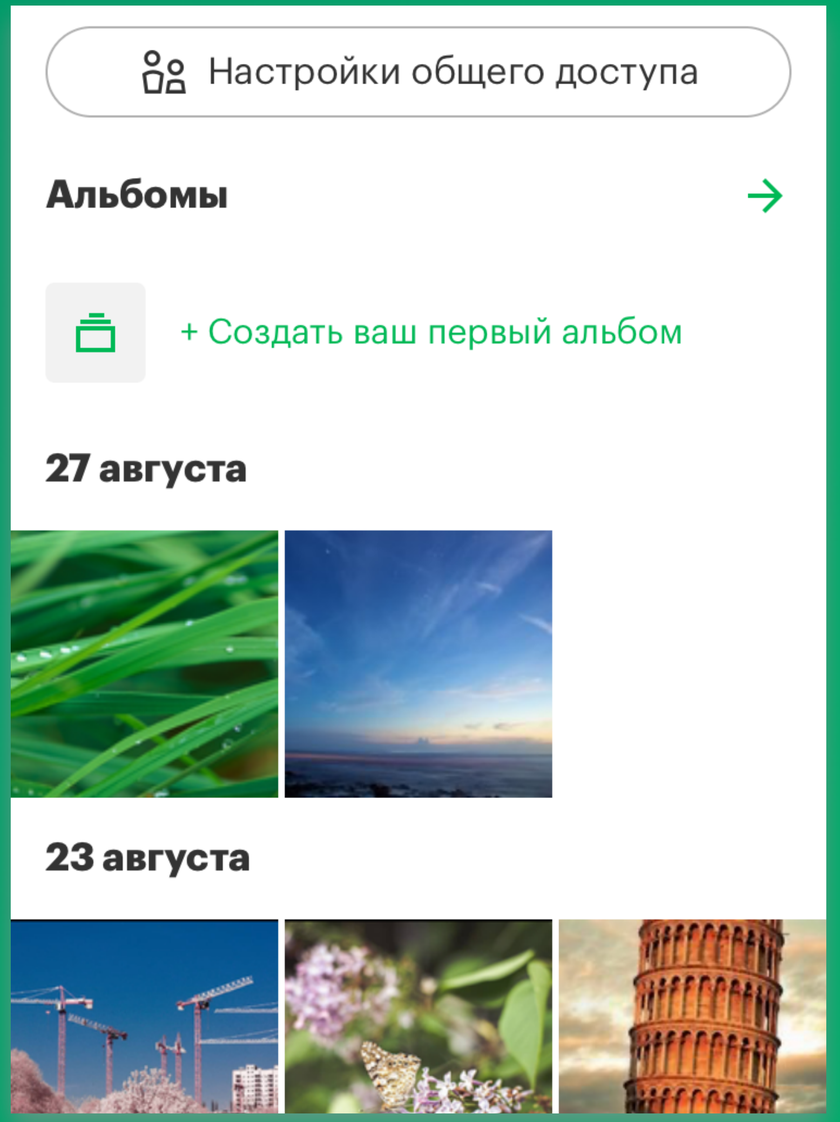
Загрузка в бекграунде

- Не загружаем видео в бекграунде
- Background URLSession + events
- Silent notifications

Выводы

- Метаданные для отредактированных фото отличаются
- Сравниваем хеши на клиенте и сервере
- Хеш для видео вычисляем “по кусочкам”
- OperationQueue удобен для параллельных загрузок

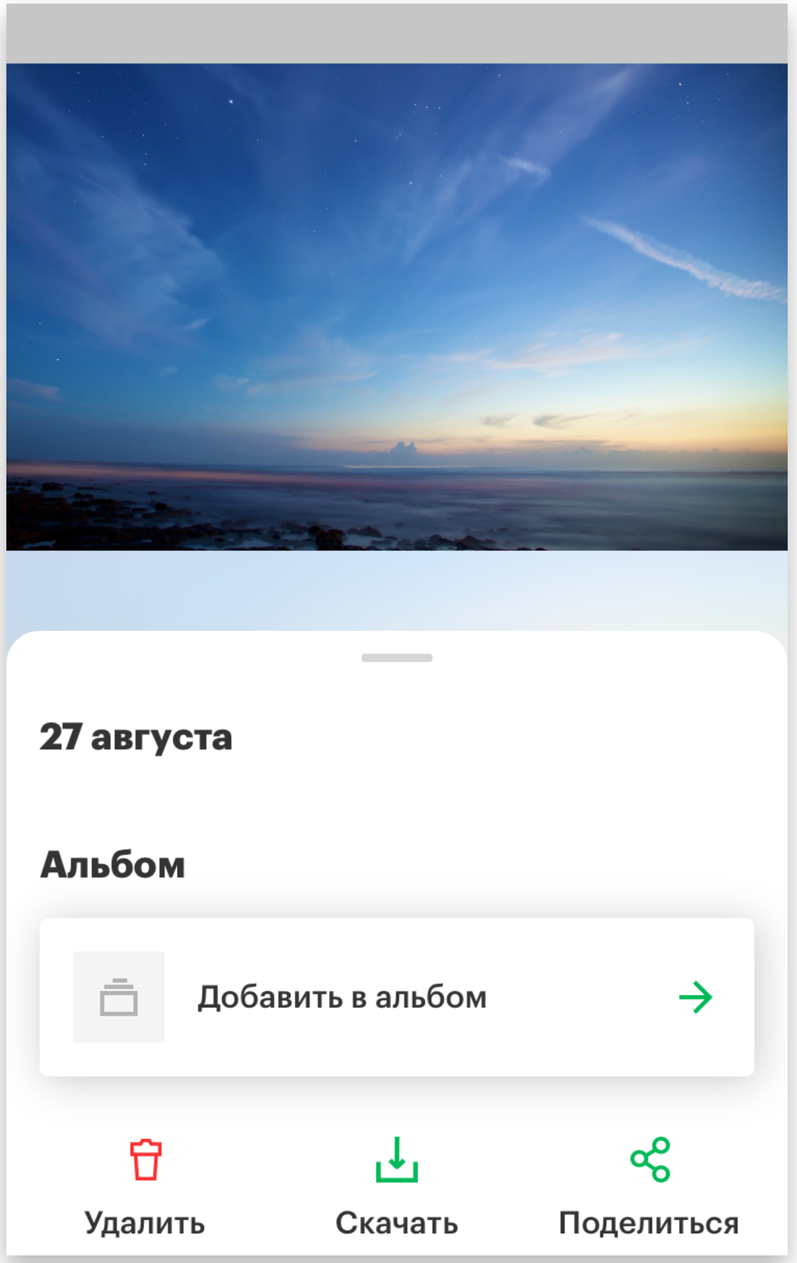
Общий диск


Настройки общего доступа
Альбомы
+ Создать ваш первый альбом
27 августа
23 августа

Общий диск

Задачи

- Идентичная логика с фото
- Идентичный (почти) UI
- Не пересекается с приватным диском



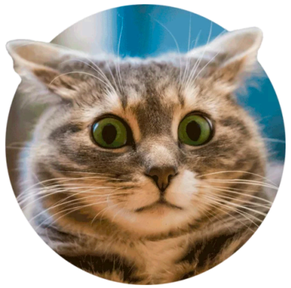
Общий диск

Разделить приватный и общий диск

```
var cfg = Realm.Configuration()
cfg.fileURL = …
let realm = try Realm(cfg)
```

Общий диск

Идентичная логика

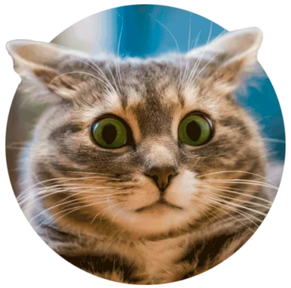


Состояния в сервисе?

```
final class PhotoService: PhotosServiceProtocol {

    init(ownerType: OwnerType) {…}
}
```

Общий диск

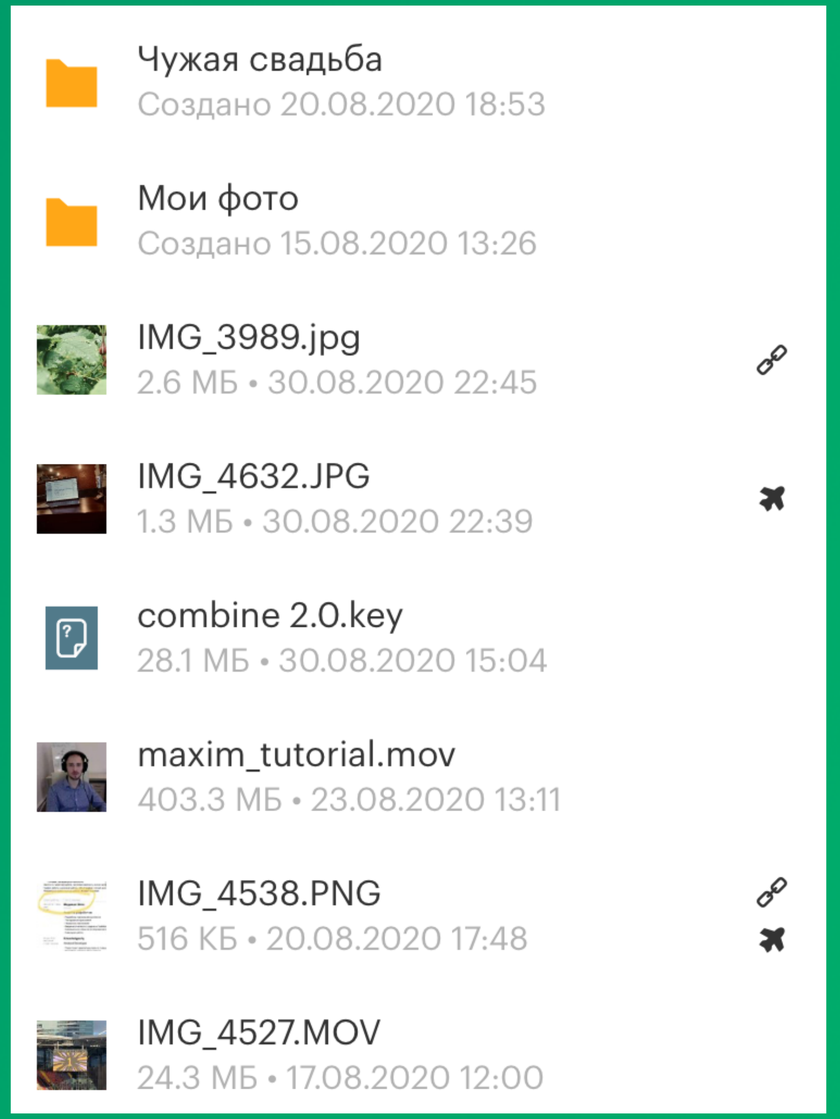
Идентичная логика



Состояния в сервисе?

- Убираем избыточный контроль зависимостей из ViewModel
- Так или иначе придется передать нужную конфигурацию БД в сервис

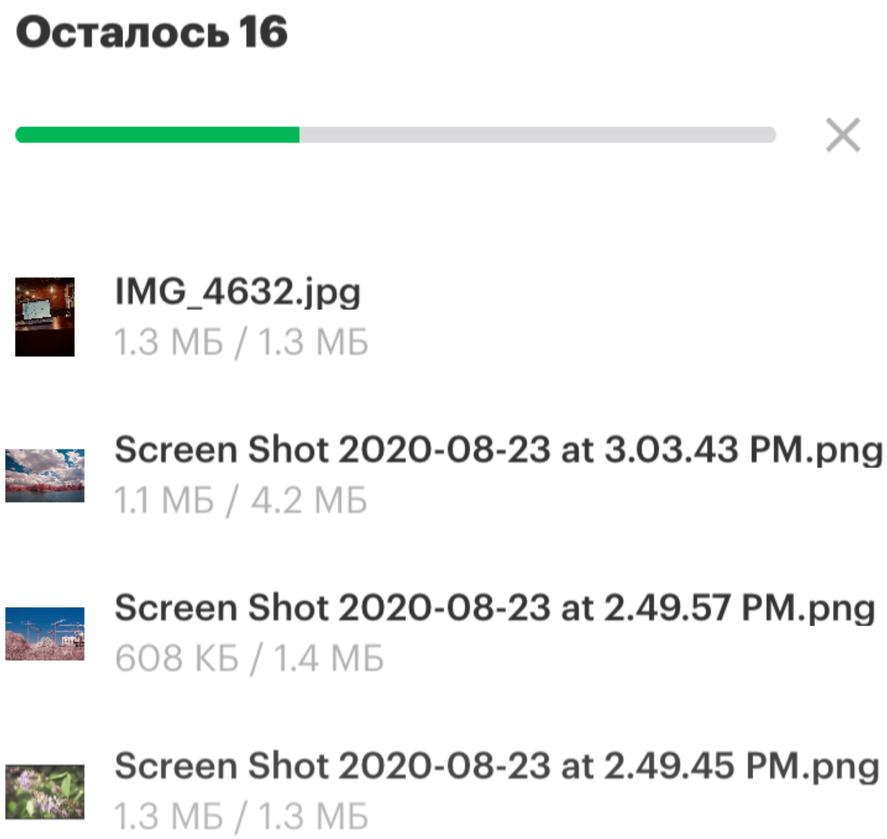
Файлы


Чужая свадьба
Создано 20.08.2020 18:53
Мои фото
Создано 15.08.2020 13:26
IMG_3989.jpg
2.6 МБ • 30.08.2020 22:45
IMG_4632.JPG
1.3 МБ • 30.08.2020 22:39
combine 2.0.key
28.1 МБ • 30.08.2020 15:04
maxim_tutorial.mov
403.3 МБ • 23.08.2020 13:11
IMG_4538.PNG
516 КБ • 20.08.2020 17:48
IMG_4527.MOV
24.3 МБ • 17.08.2020 12:00

Загрузка файлов

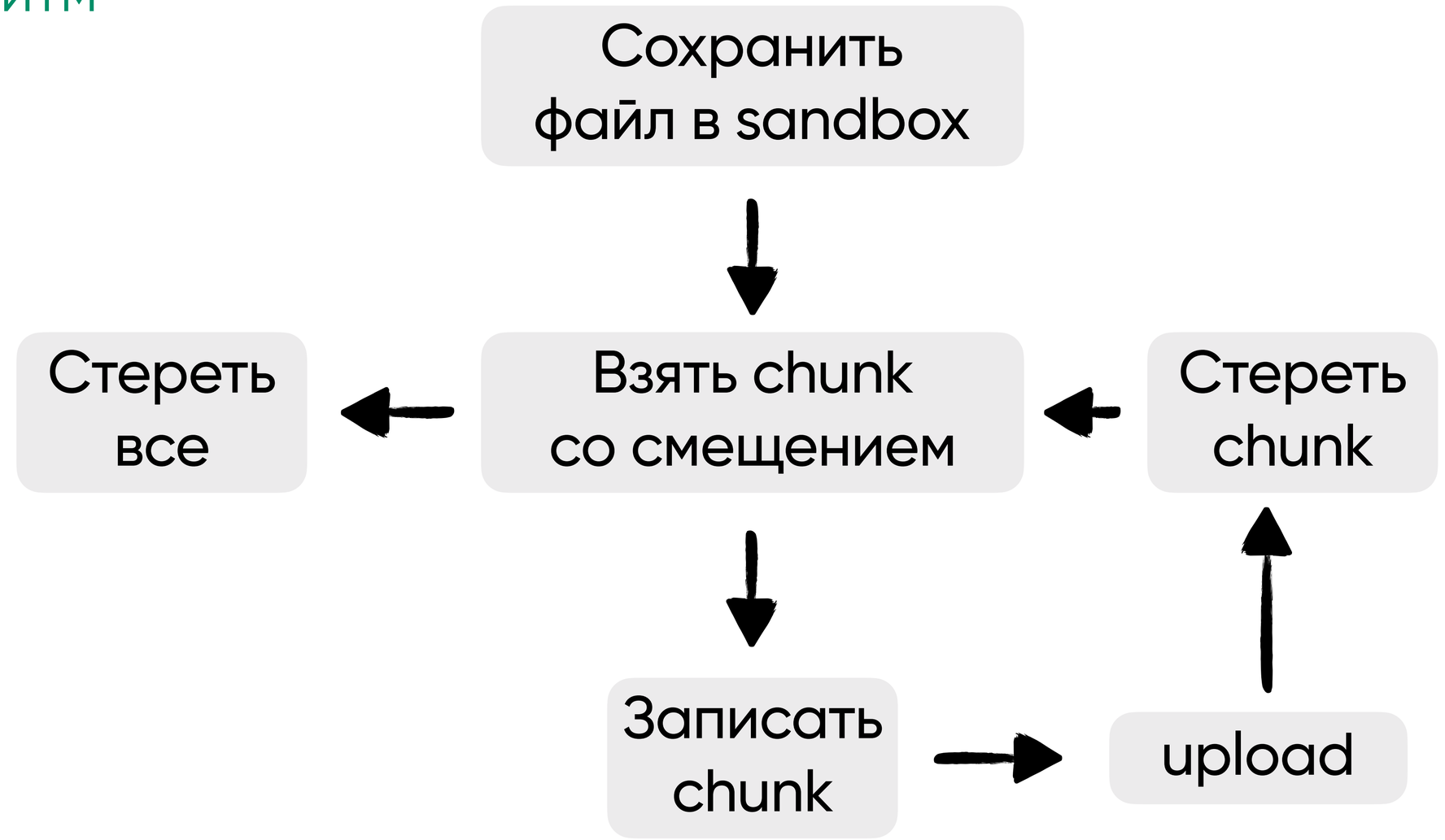
Задачи

- Загрузка
- Просмотр
- Extensions



Загрузка больших файлов

Алгоритм



Загрузка файлов

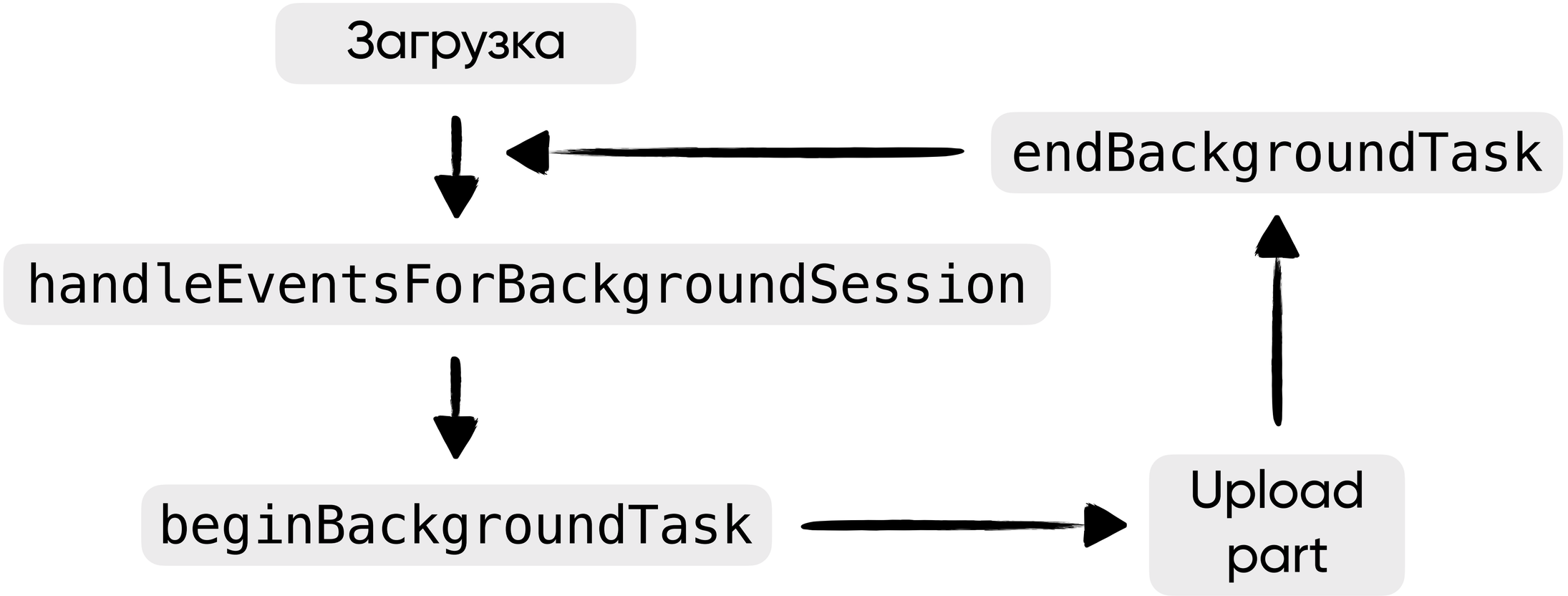
Алгоритм

Целый файл	С разбиением
~78сек.	~82сек.

93 мб, WiFi, foreground

Фоновая загрузка

Продлеваем жизнь



iOS14 

Загрузка файлов

Default vs Background URLSession

Default	Background
~82с.	~88с.

93 мб, WiFi, foreground

Просмотр файлов

Форматы файлов

Этап 1	QuickLook
Этап 2	Фото/видео
Этап 3	PDF
Этап 4	Аудио

Просмотр файлов

Аудио

- [ ] gsm
- [x] amr
- [x] aiff
- [x] aac
- [ ] ogg
- [x] wav
- [ ] mid
- [x] flac
- [x] mp3
- [x] m4a
- [ ] wma

Просмотр файлов

Видео

- [x] mp4
- [x] mov
- [ ] mkv
- [ ] 3gpp
- [ ] wmv
- [x] flv
- [ ] webm

Просмотр файлов

Медиа

QuickLook → 

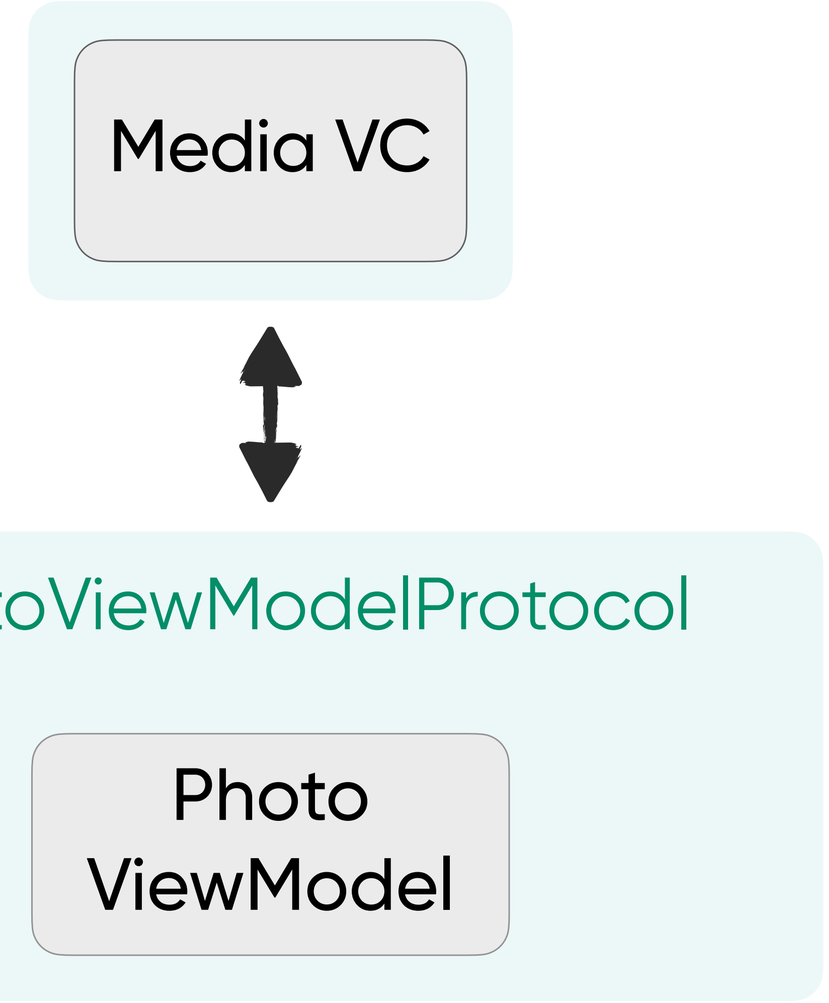
Просмотр файлов

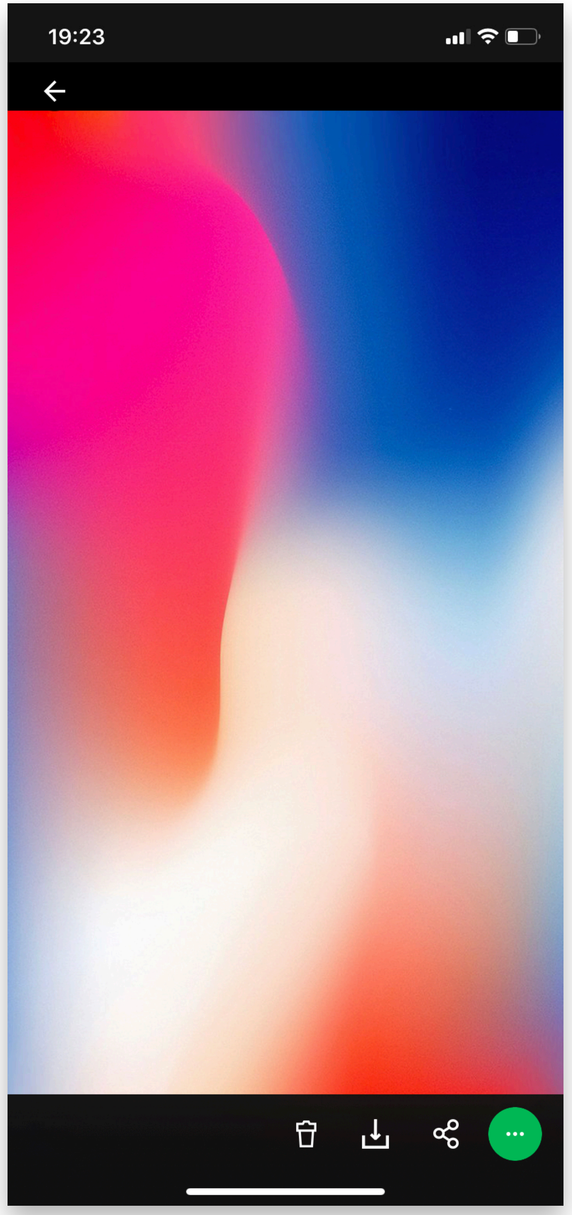
PDF

QuickLook → PDFView

Просмотр файлов

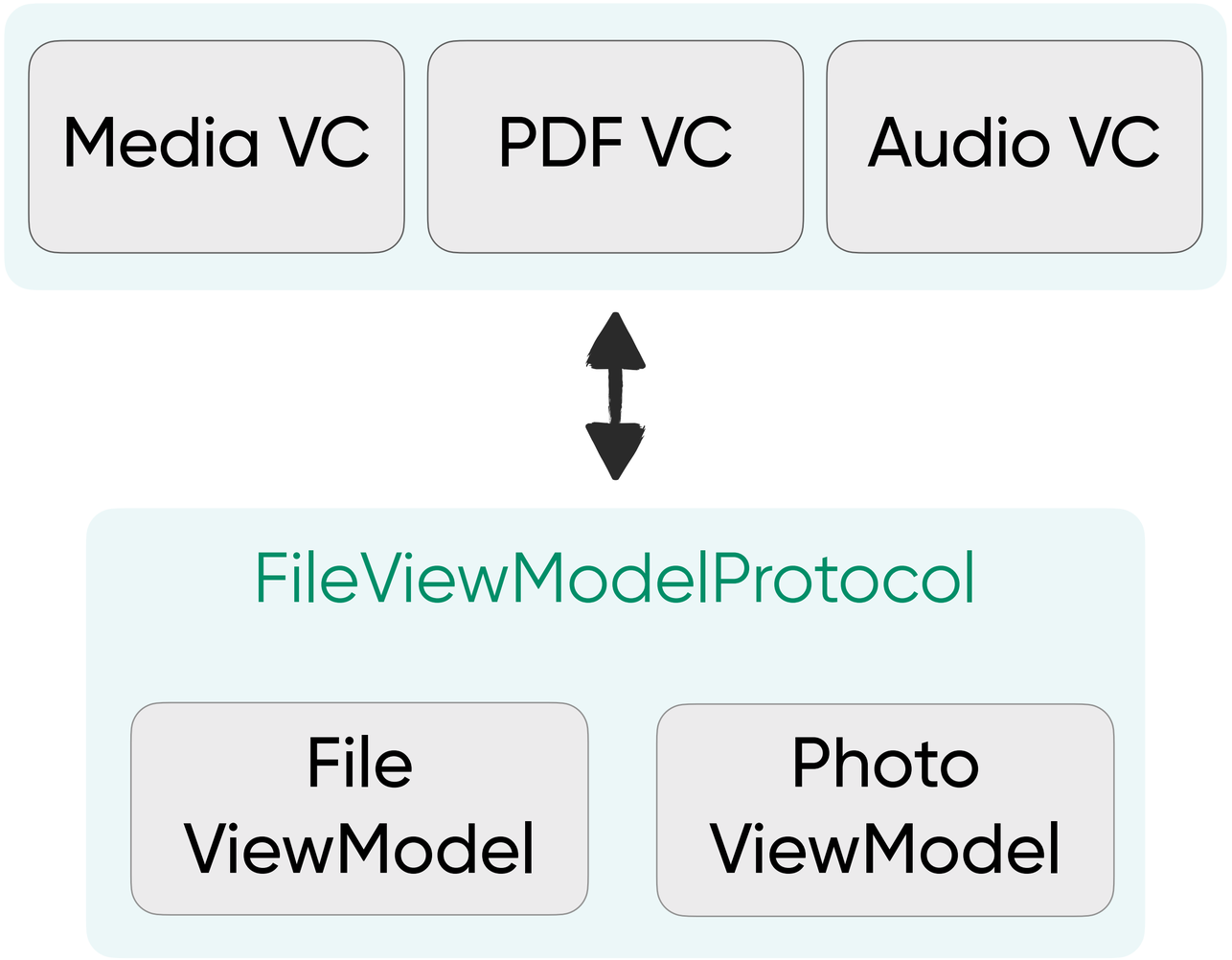
MVVM ❤️

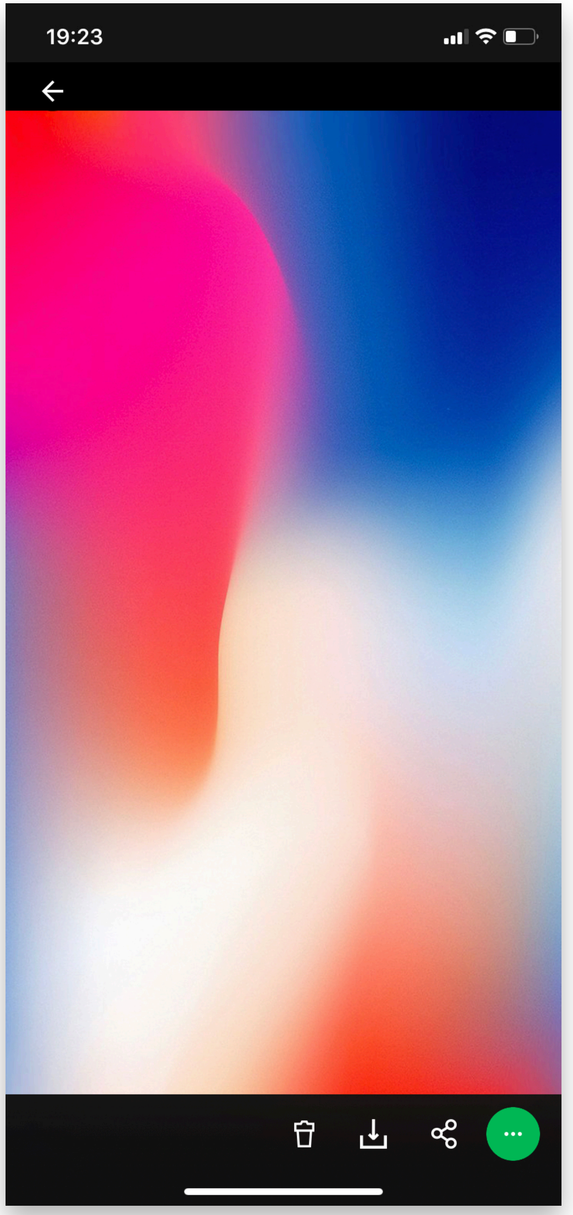




Просмотр файлов

MVVM ❤️

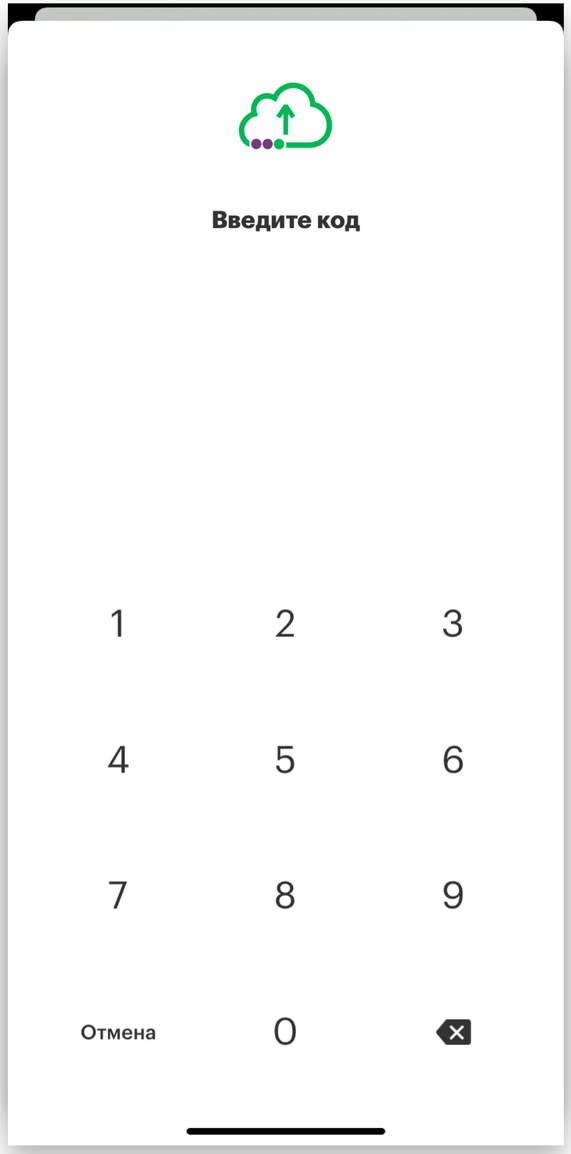




Share extension; FileProvider extension

Задачи

- Совершать авторизованные вопросы
- Отображать состояния авторизован / нет
- Защита кодом
- **Проверить код на модульность**



Офлайн режим

Файл можно...

- Удалить
- Переместить
- Скопировать

... с другого устройства

Офлайн режим

Связь с метаданными

- Файл на диске != файл в БД
- `file.offlinePath = “/mobius_preza.key”`
- Следить, чтобы файл не перезатерался

Следим на местом на диске

Стратегии очистки

- Разделяем по папкам офлайн и превью
- Чистим кеш файлов в `applicationWillTerminate`
- Ограничения размера кеша картинок
- `didReceiveMemoryWarning*`

Выводы

Файлы

- Фоновая загрузка это сложно(
- Синхронизировать метаданные с файлами на диске
- Пишите тесты :)

Что в итоге

Основные фокусы

- Загрузка просмотр фото и файлов
- Автозагрузка
- Дополнительные фишки
- МНОГО ХАЛЯВНОГО МЕСТА

Directed by
MegaFon & AGIMA

email: i.vedeneev@agima.ru
telegram: @getmaxx