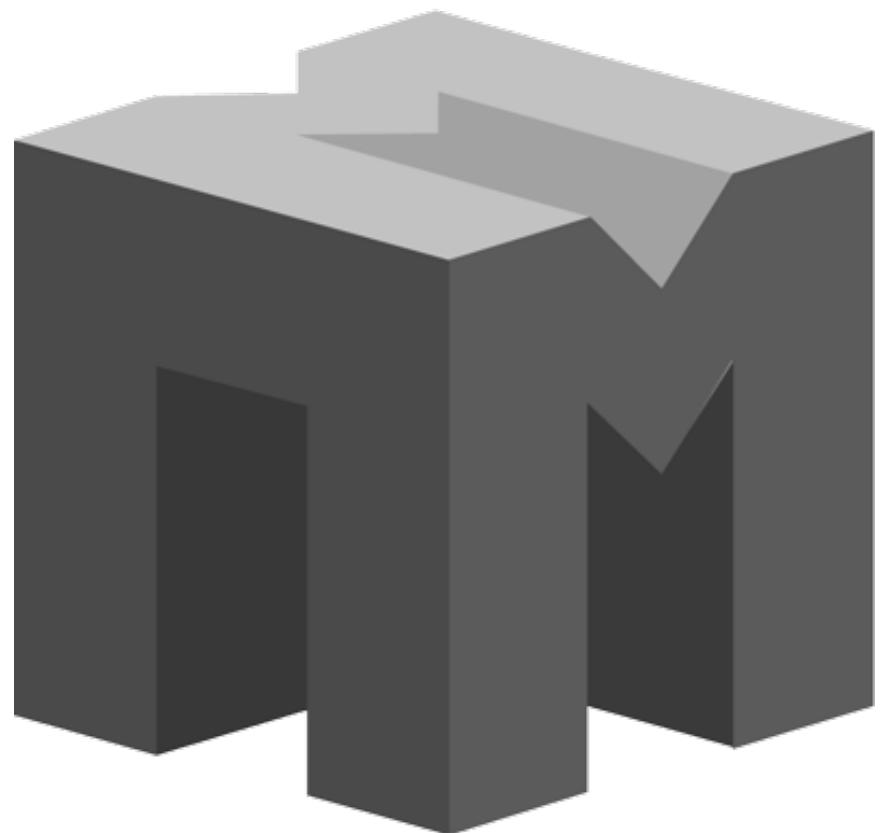


E D U C
S M I F
N O I T A

iOS Школа



E D U C A T I O N

Александр Филимонов

Работа с данными

Классификация данных

Получение данных

Отправка данных

Обработка данных

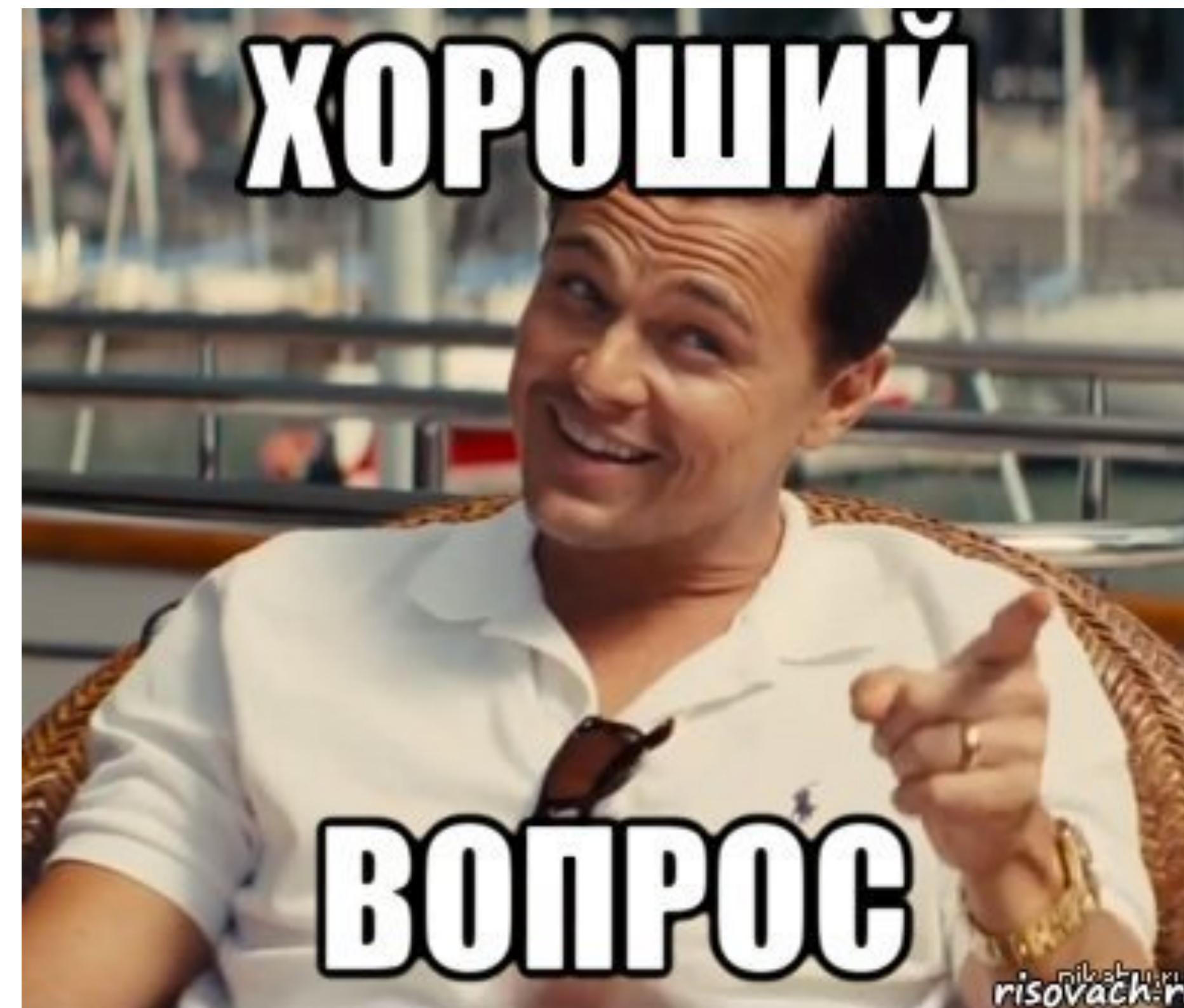
Классификация данных

Получение данных

Отправка данных

Обработка данных

Что такое данные???



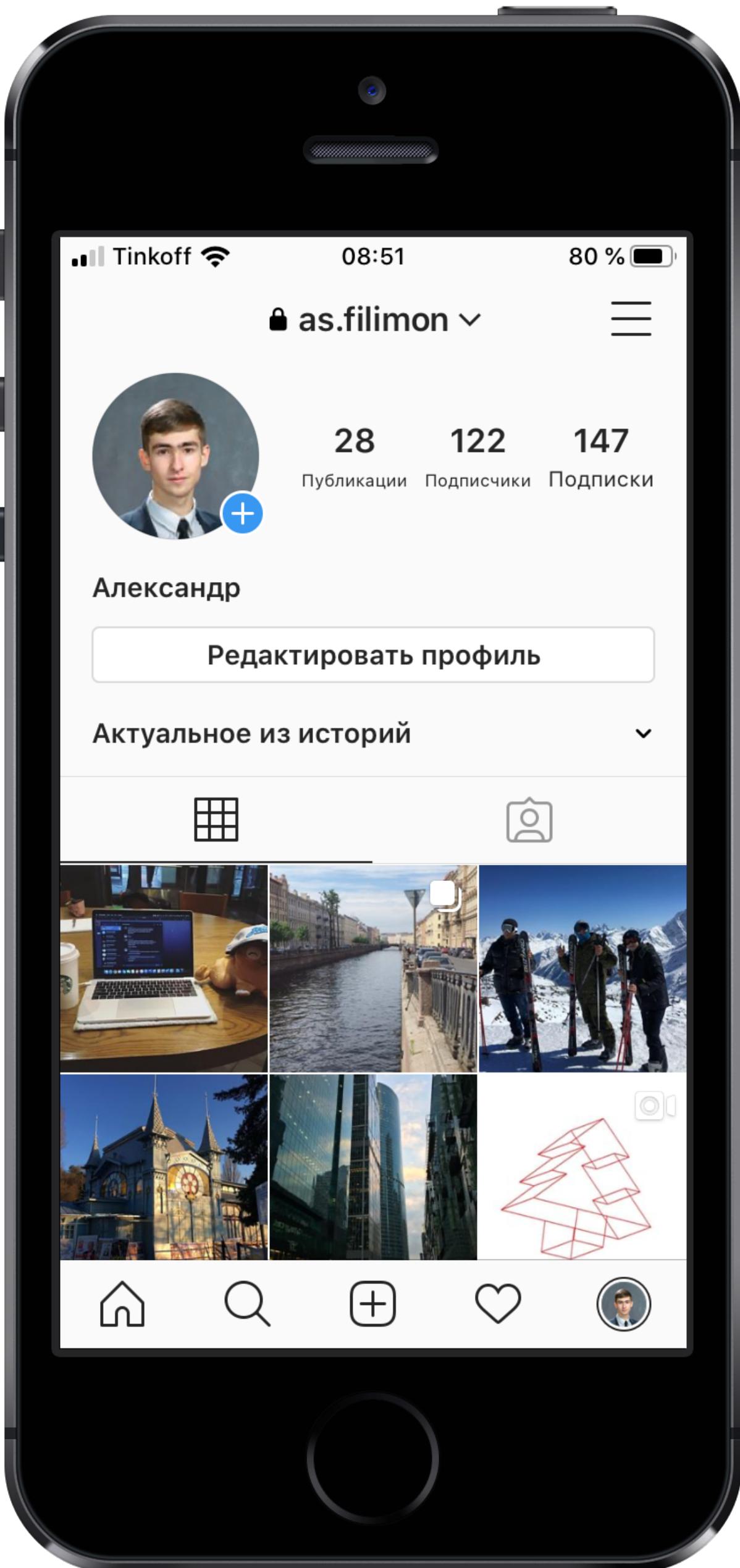
Вот это данные?



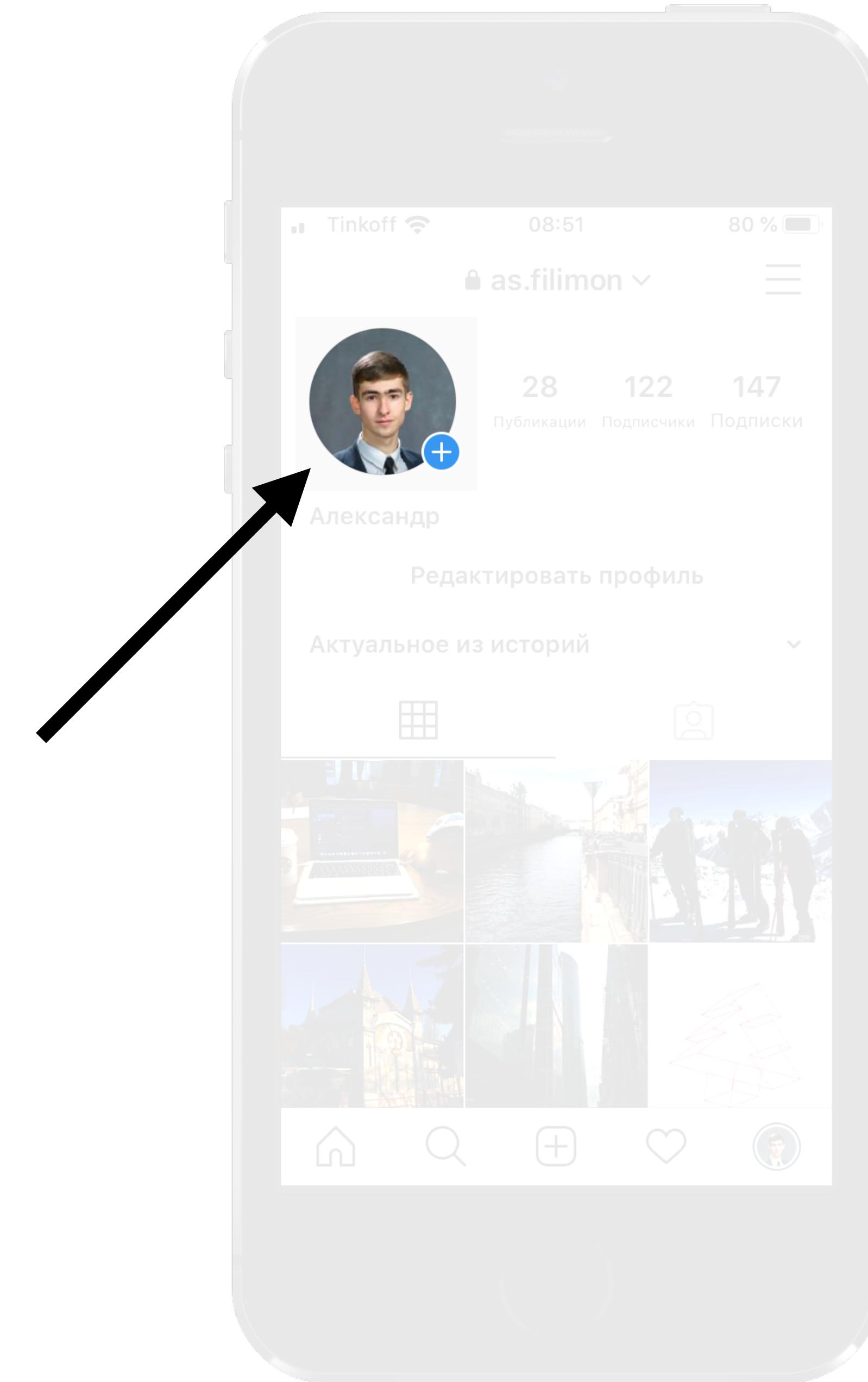
Вот это данные?



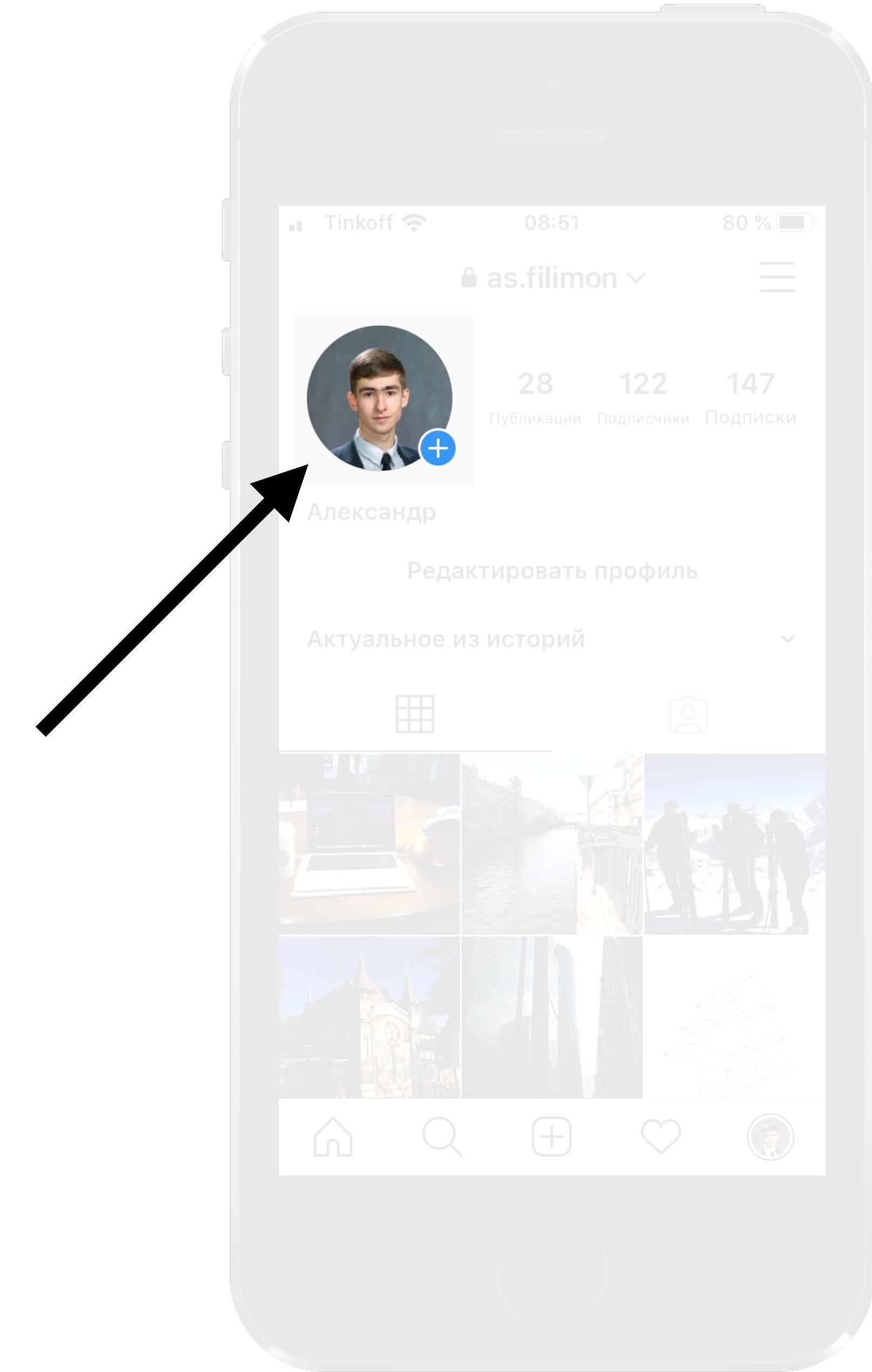
Обычное приложение



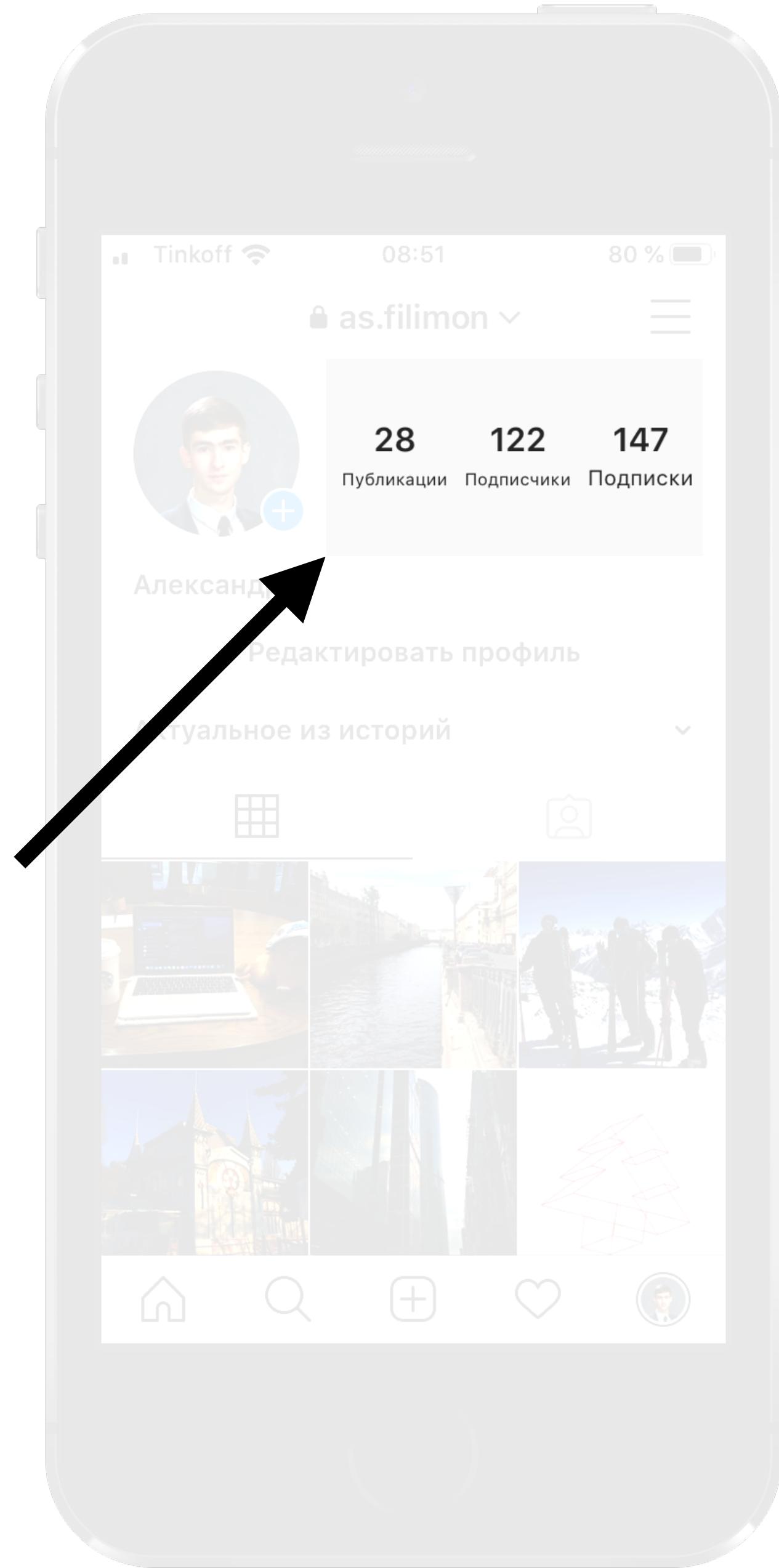
Вот это данные?



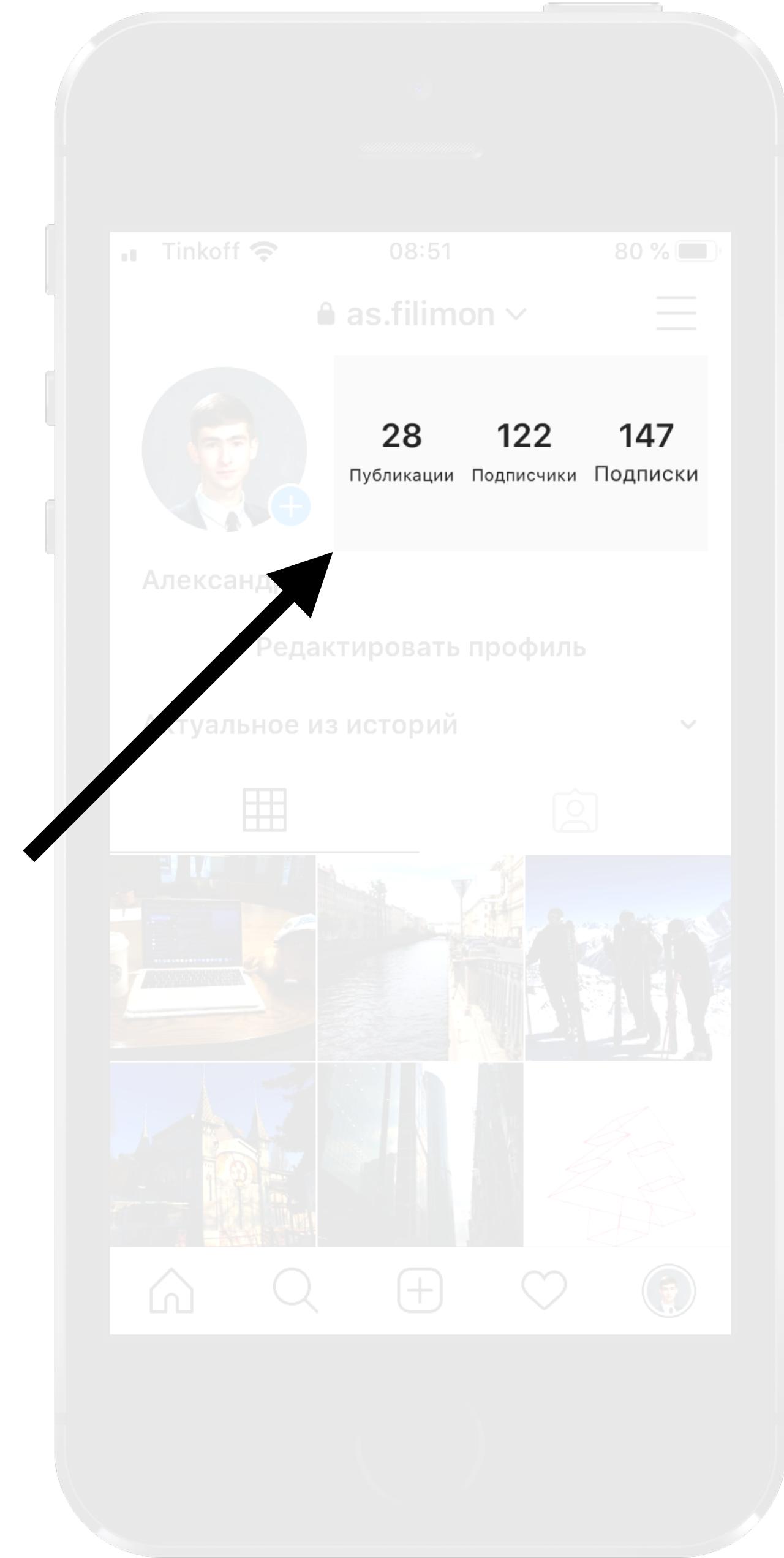
Вот это данные?



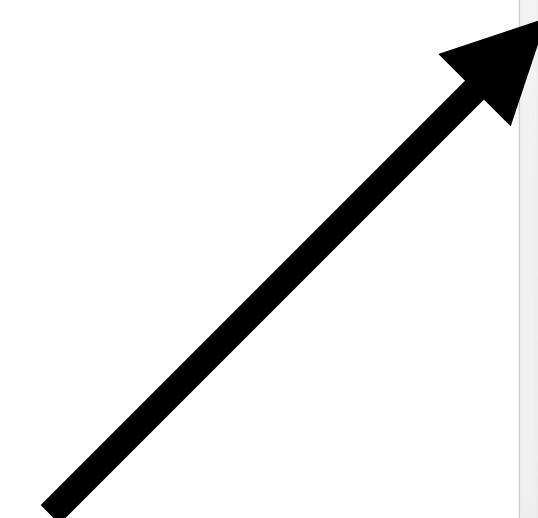
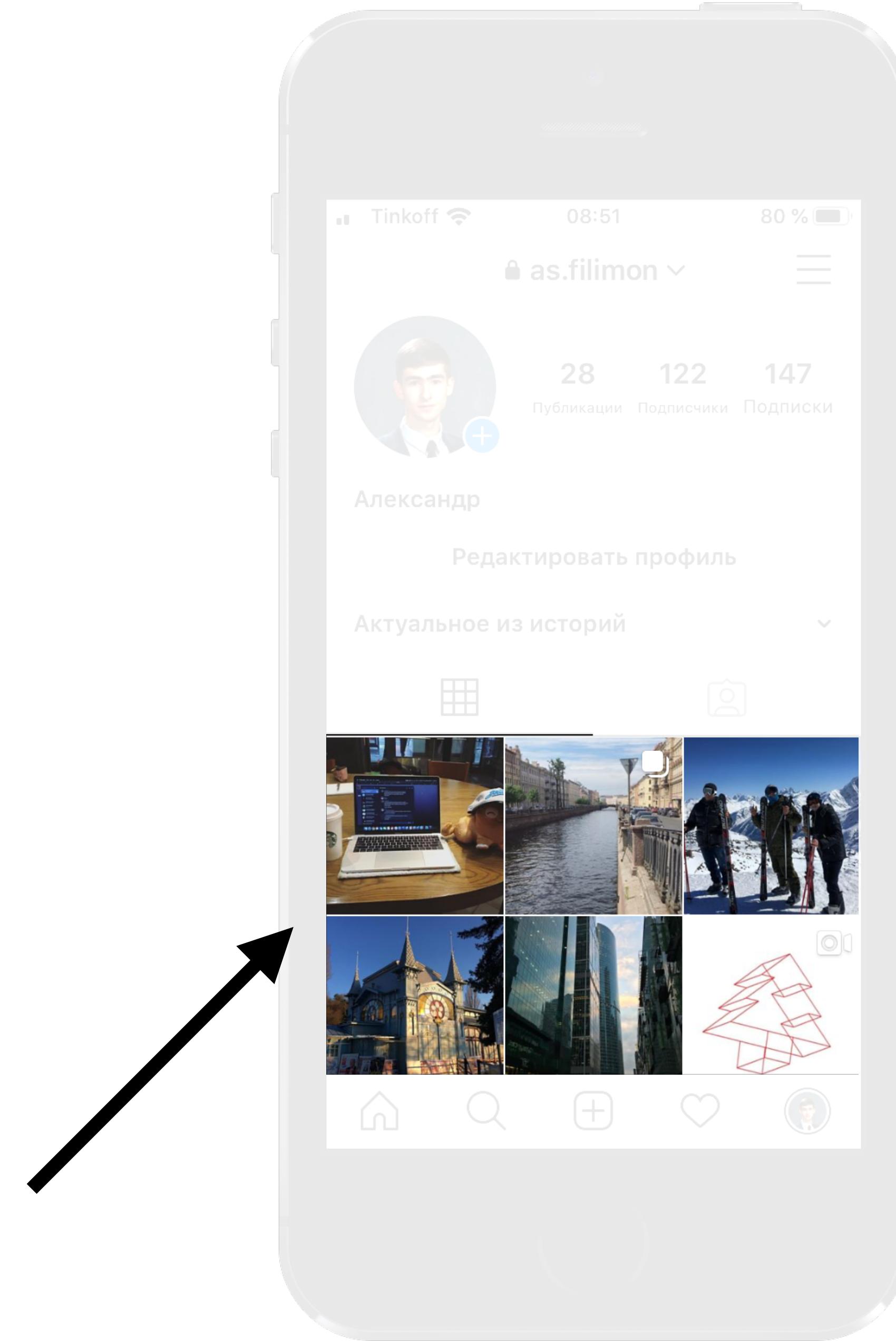
Вот это данные?



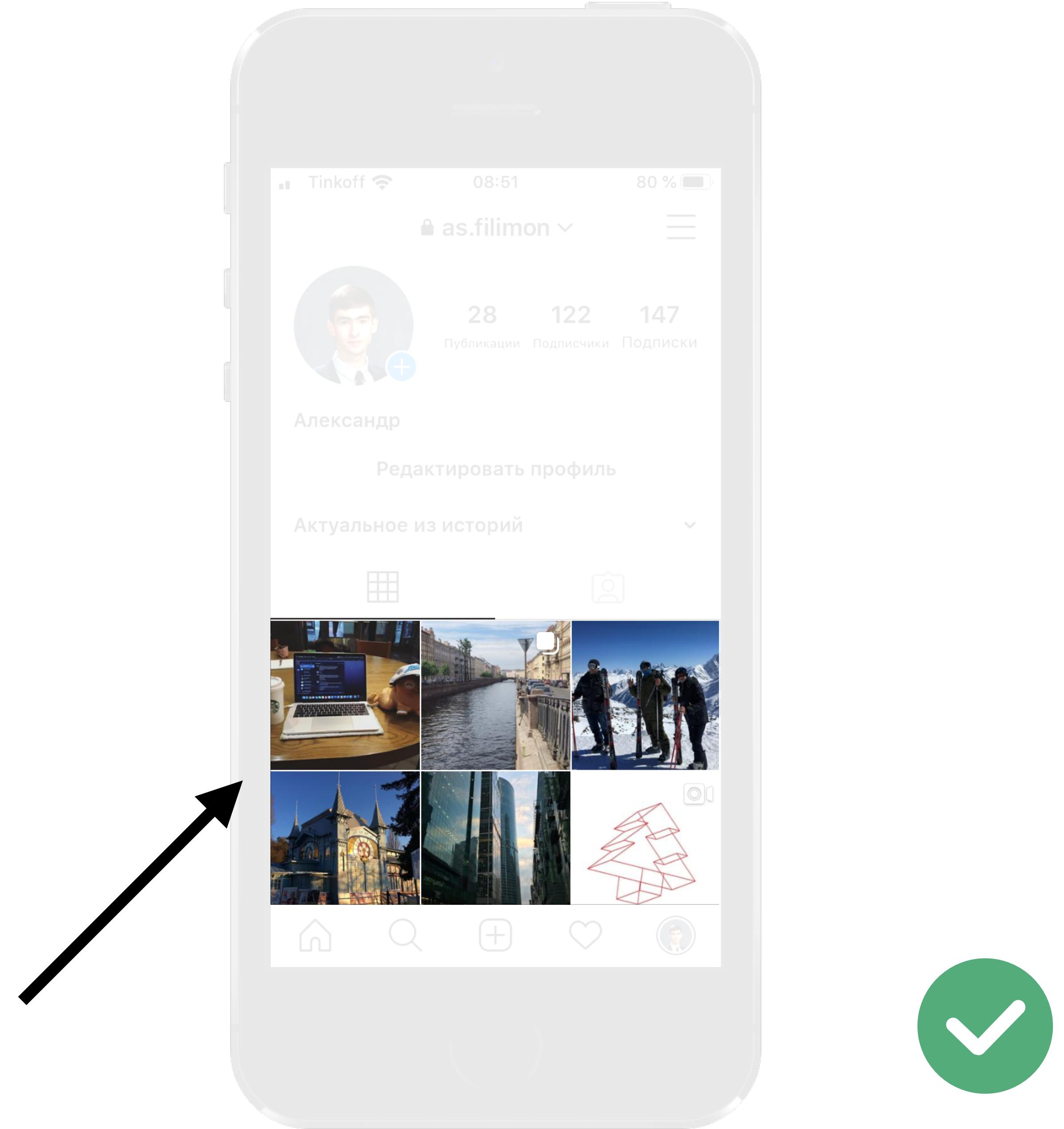
Вот это данные?



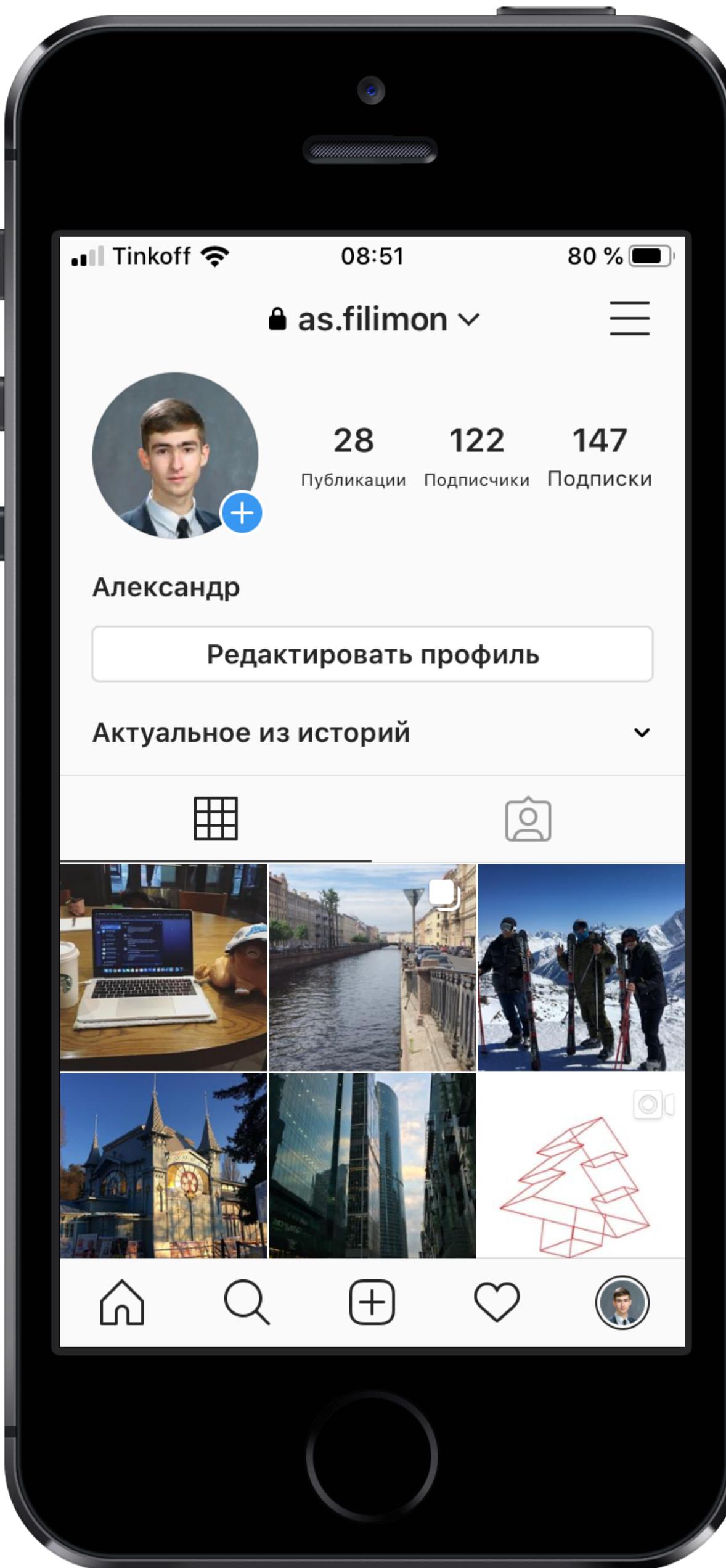
Вот это данные?



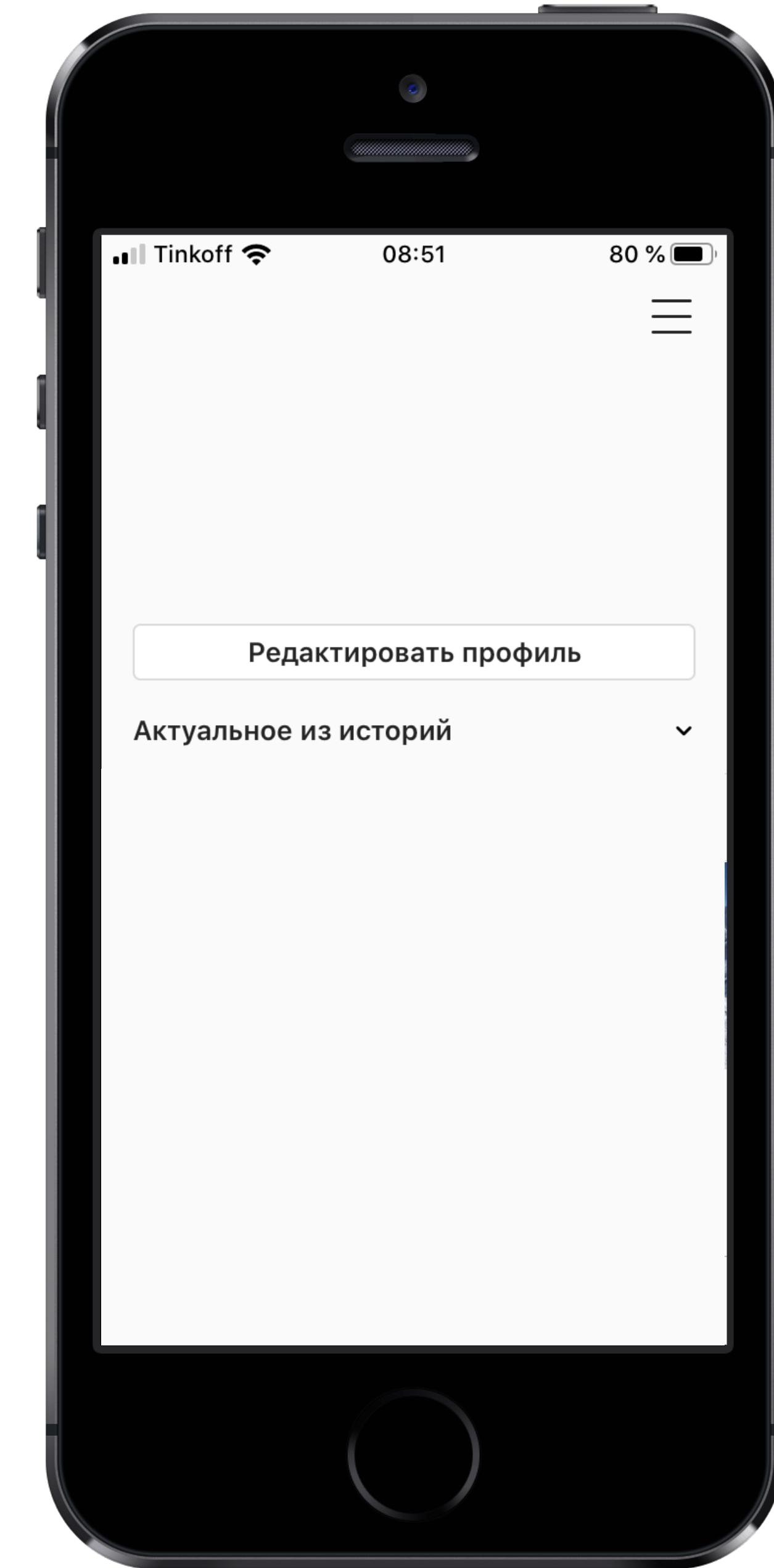
Вот это данные?



Обычное приложение



Приложение без данных



А это данные?

```
{  
    "login": "surf2019",  
    "password": "12345678"  
}
```

А это данные?

```
{  
    "login": "surf2019",  
    "password": "12345678"  
}
```



А это данные?

<https://api.unsplash.com/photos/random>

А это данные?

<https://api.unsplash.com/photos/random>



Что такое данные

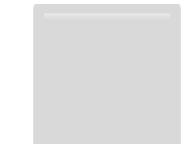
Что такое данные

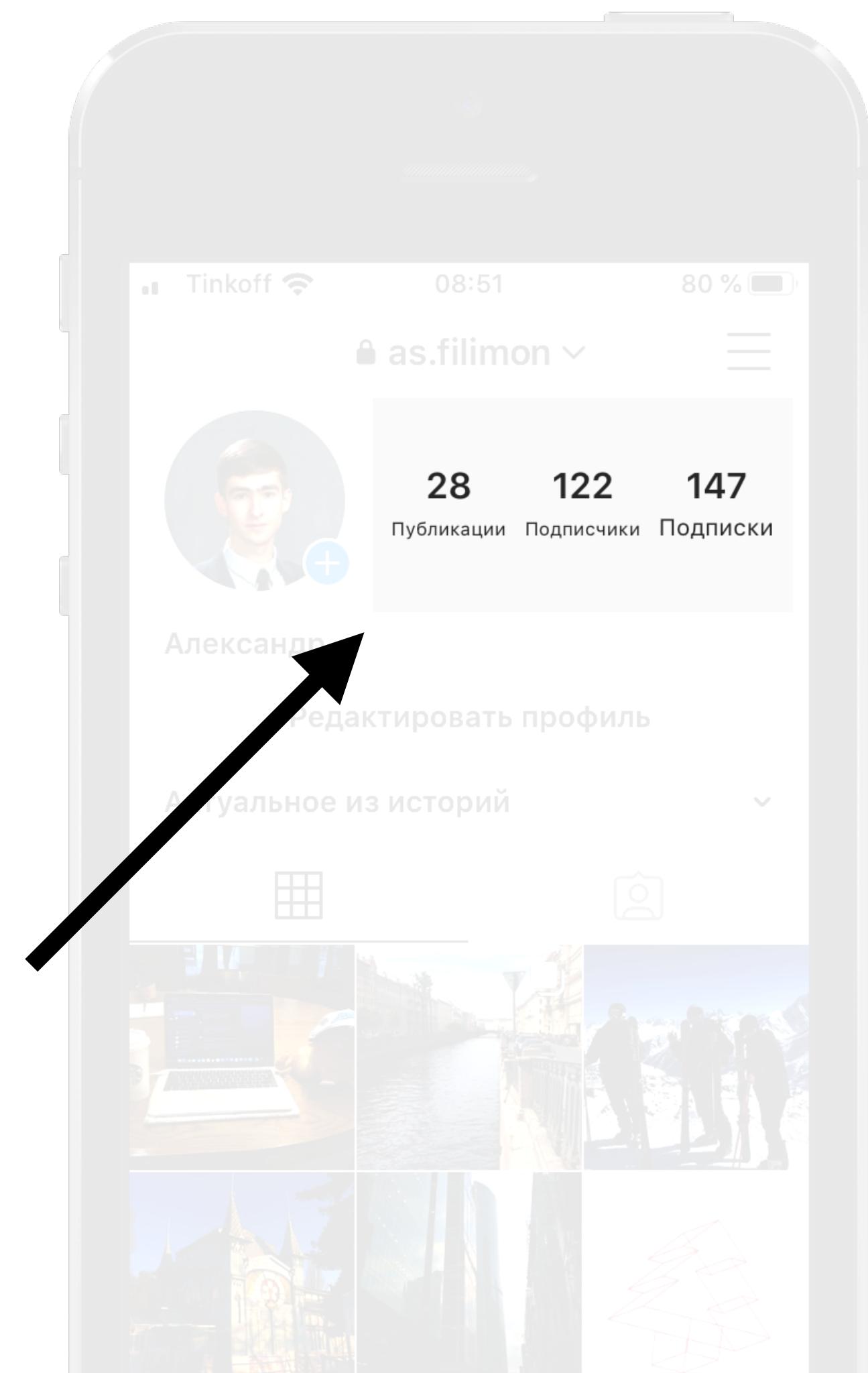
**Все, что наполняет приложение, и даже чуточку
больше!**

Классификация данных по виду

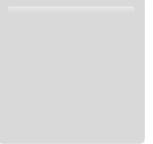
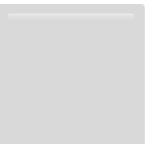
-  **Контент**
(товары, книги, тексты, ...)
-  **Медиа**
(картинки, gif, музыка, ...)
-  **Приватные данные**
(пароли, номера карт, ...)
-  **Служебные данные**
(настройки, адреса серверов, ...)

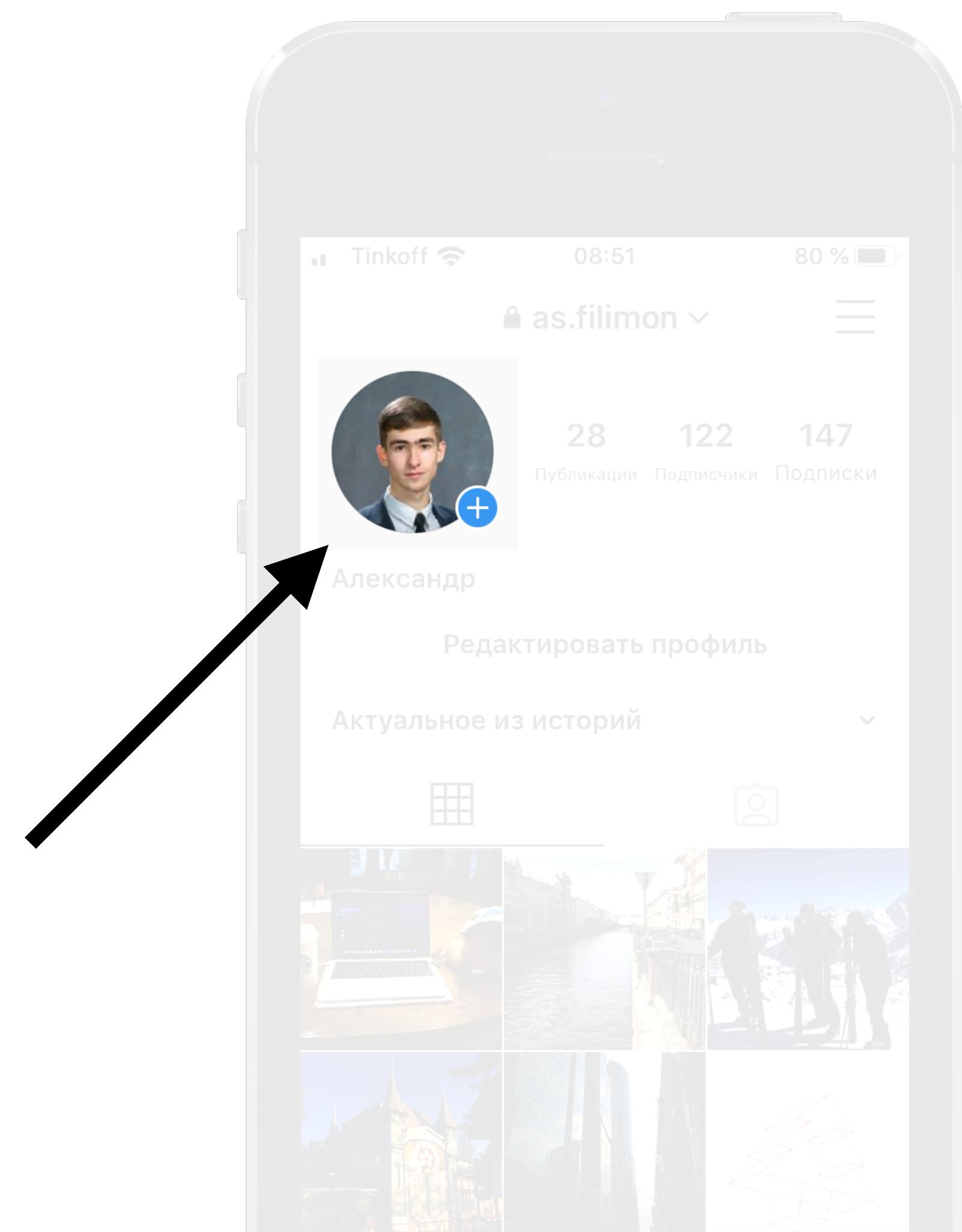
Классификация данных по виду

-  **Контент**
(товары, книги, тексты, ...)
-  **Медиа**
(картинки, gif, музыка, ...)
-  **Приватные данные**
(пароли, номера карт, ...)
-  **Служебные данные**
(настройки, адреса серверов, ...)

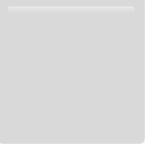


Классификация данных по виду

-  **Контент**
(товары, книги, тексты, ...)
-  **Медиа**
(картинки, gif, музыка, ...)
-  **Приватные данные**
(пароли, номера карт, ...)
-  **Служебные данные**
(настройки, адреса серверов, ...)



Классификация данных по виду

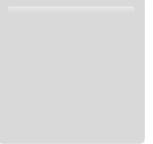
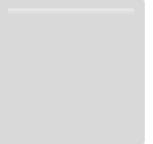
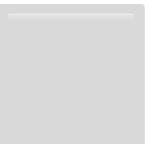
-  **Контент**
(товары, книги, тексты, ...)
-  **Медиа**
(картинки, gif, музыка, ...)
-  **Приватные данные**
(пароли, номера карт, ...)
-  **Служебные данные**
(настройки, адреса серверов, ...)

{

}

```
"login": "surf2019",  
"password": "12345678"
```

Классификация данных по виду

-  **Контент**
(товары, книги, тексты, ...)
 -  **Медиа**
(картинки, gif, музыка, ...)
 -  **Приватные данные**
(пароли, номера карт, ...)
 -  **Служебные данные**
(настройки, адреса серверов, ...)
- [https://
api.unsplash.com/
photos/random](https://api.unsplash.com/photos/random)

Классификация данных по формату

-  **Бинарные**
 - Картинки
 - Видео
 - ...
-  **Текстовые**
 - JSON
 - XML
 - ...

Классификация данных по формату

- **0 1** Бинарные

- Картинки
- Видео
- ...

gірНЦ&9Cсн»4ОЦАп‘≥Кщ9РGчфоБХАв` ЙИ'А8в1≥T
ЦІЇ/хюSЩ5-...гeM` M.j!ЗЛе“|ёЄ€g9юо)»W
ырАН&TE'ТМrЖиїХ7+FEічI≤vv!6ДшЭХ†гЗ·K—){‘Ь/
зjъ&fo^x
fжGJтгвмъЇlq\$b€B–
q7Y;wЙPC§eЇЧЌJМðў2[;JЩ'SыUўЯRд.∞мъJDBo0€
лRQrЪ&fMZ,NHГХb¶</)FcГўХK8q√62ЕГ*~;ИДT®O
?6#[Щ®μЧ»3у≈ЦИfNº8-Ѓ-%GkyuuукшM
nJУ)√<ВщФўб\¬рMoЎe°\$Eы/¶ФЕ%јzНъФж'hнq™0?
J·JБў\љьўэлS©б«гSE ...ъ8cђ\ёfM<x
ИХ@[ШrБ€≥ѓию“MDРБ nZЌрM4xИ>У@)ш"!РПћ
SП

- **■** Текстовые

- JSON
- XML
- ...

Классификация данных по формату

-   Бинарные

- Картинки
- Видео
- ...

gipHЦ&9Cсн»4OЦAn'≥Kш9PGчoБХAv`НЬи'A8v1≥T
ЦIћ/xгюSЩ5-...гeM`M.j!ЗLe“|ё€€g9юo)»W
ыpAн&TE'тMгЖuїX7+FEiчl≤vв!6ДшЭХtг3·K—){‘b/
zjь&fo^x
fжGJtгbмъђlq\$b€B-
q7Y;wЙPCњеђЧKJМДў2[;JЩ'SyIуЯRд·∞мъJDBo0€
лRQrъ&fMZ,NHГХb¶</)FcГўХK8q√62ЕГ*~;ИДT®0
?6#[Щ®μЧ»Зу≈ЦИfNº8-Ѓ-%GkyuuukшM
nJУ)√<ВщФўб\¬рMoЎe°\$Eы/¶ФE%јzНЬФж'hнq™0?
J·JБУ\љьбЭЛS©б«гSE ...ъ8cђ\ёfM<x
ИХ@[ШrБ€≥ѓию“MDPБ nZKрM4xИ>У@)ш"!РПћ
SП

-  Текстовые

- JSON
- XML
- ...

Классификация данных по формату

- **0 1** Бинарные
 - Картинки
 - Видео
 - ...
- **■** Текстовые
 - JSON
 - XML
 - ...



@HACKSAWACADEMY

Классификация данных по формату

-   Бинарные
 - Картинки
 - Видео
 - ...
 -  Текстовые
 - JSON
 - XML
 - ...
- ```
{
 "login": "surf2019",
 "password": "12345678"
}
```

# Классификация данных по формату

- ■ Бинарные

- Картинки
- Видео
- ...

-  Текстовые

- JSON

- XML

- ...

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC
<plist version="1.0">
<dict>
 <key>quiz</key>
 <dict>
 <key>question</key>
 <array>
 <dict>
 <key>text</key>
 <string>Вопрос</string>
 <key>answer</key>
 <string>Ответ на вопрос</string>
 </dict>
 </array>
 </dict>
</dict>
</plist>
```

# Классификация данных по формату

-  **Бинарные**

- Картинки

- Видео

- ...

-  **Текстовые**

- JSON

- XML

- ...

# **Классификация данных**

Получение данных

Отправка данных

Обработка данных

**Классификация данных**

**Получение данных**

**Отправка данных**

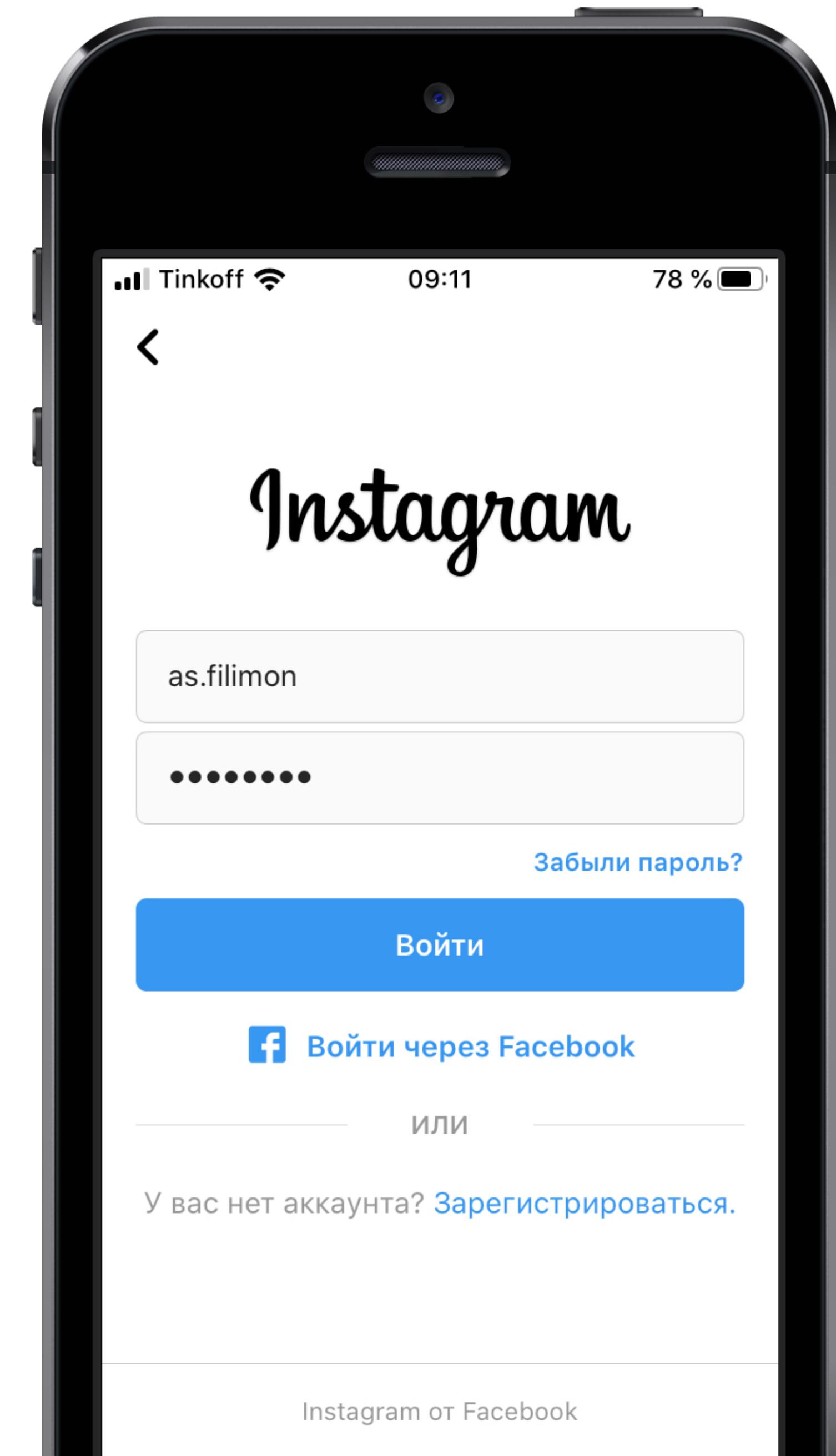
**Обработка данных**

# Откуда получать

- 😊 Пользователь
- 🌎 Интернет
- 📱 Устройство
- 📬 Входящие данные

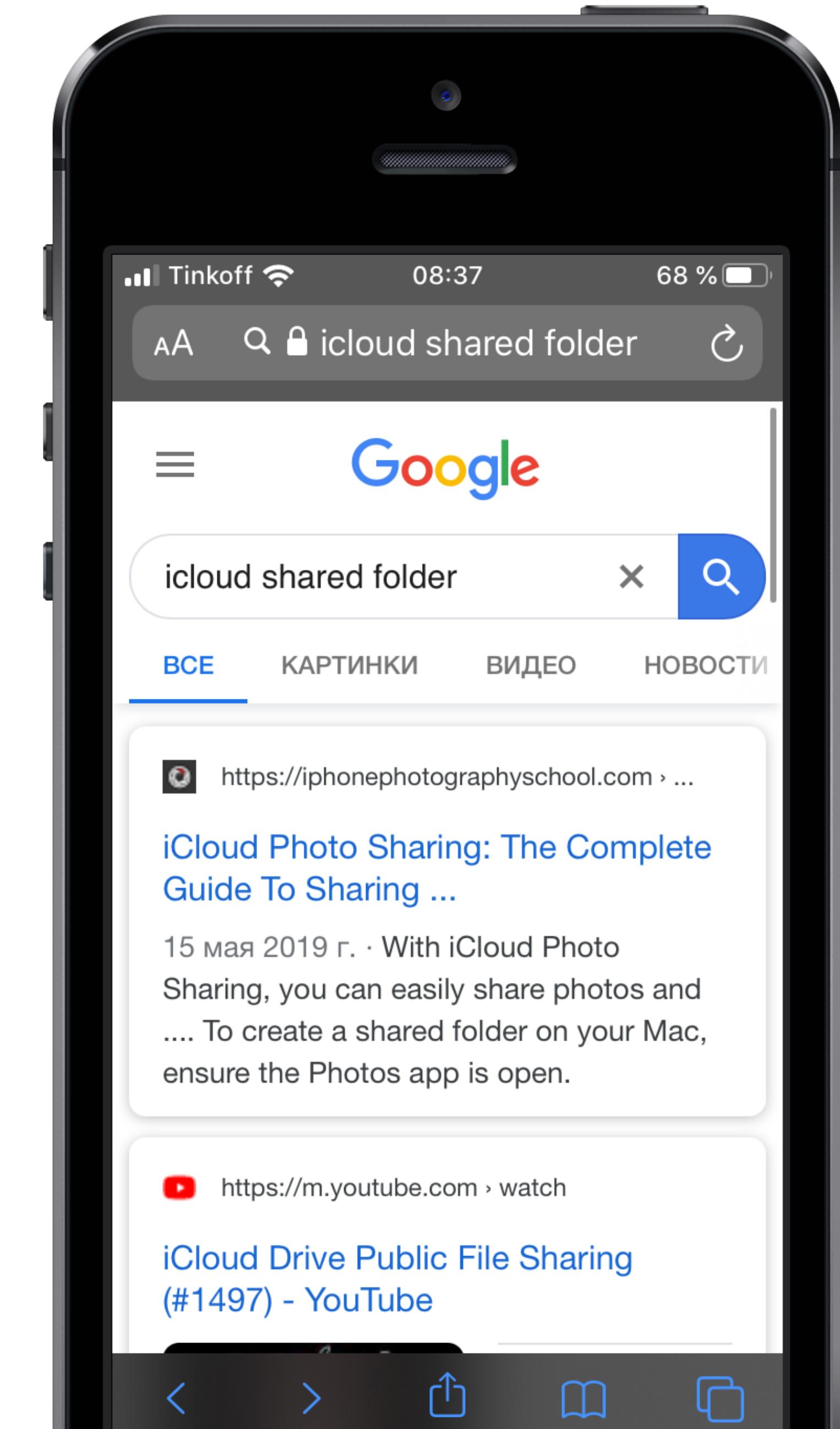
# Откуда получать

- 😊 Пользователь
- 🌎 Интернет
- 📱 Устройство
- 📬 Входящие данные



# Откуда получать

- 😊 Пользователь
- 🌎 Интернет
- 📱 Устройство
- 📬 Входящие данные



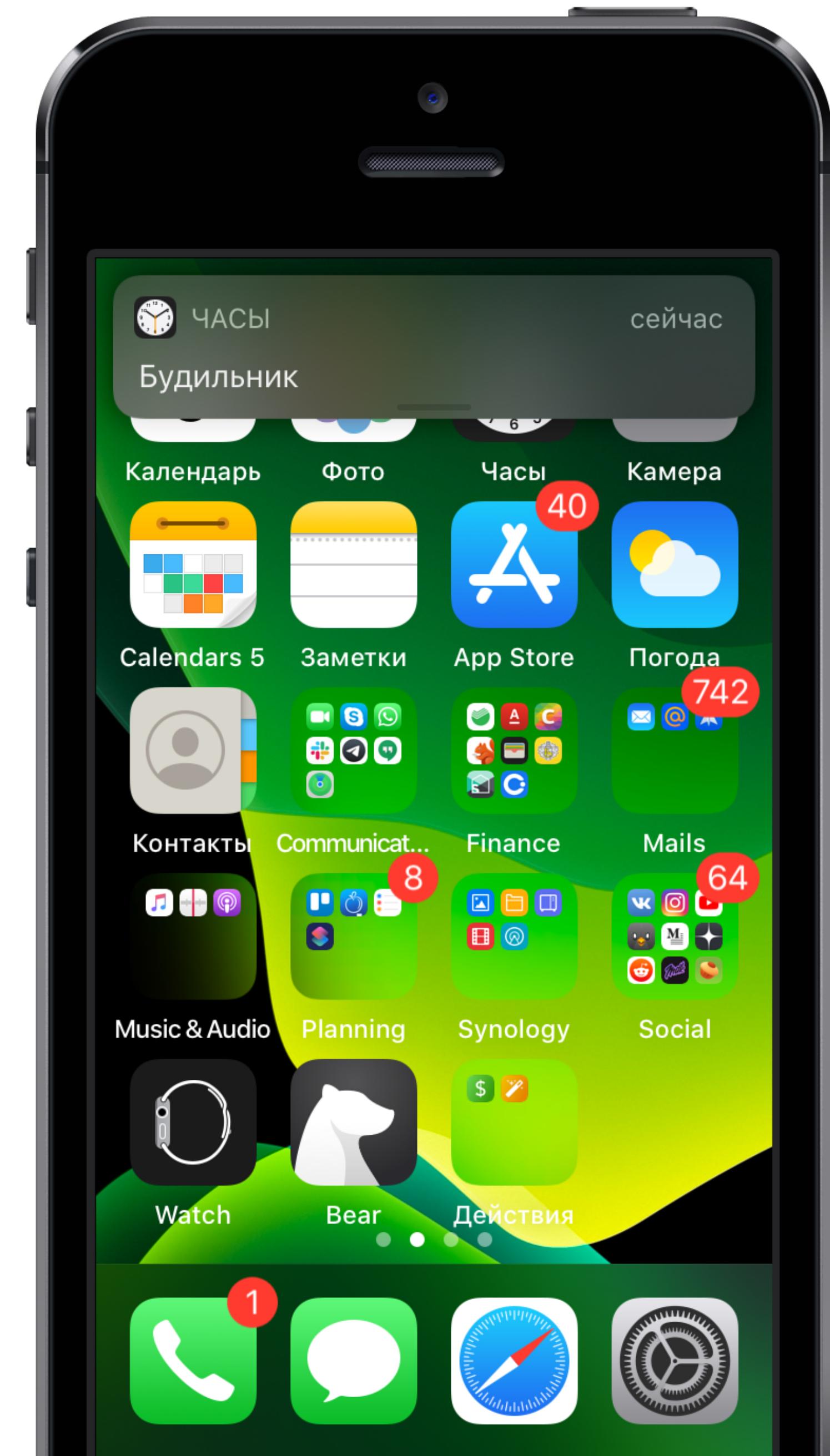
# Откуда получать

- 😊 Пользователь
- 🌎 Интернет
- 📱 Устройство
- 📬 Входящие данные



# Откуда получать

- 😊 Пользователь
- 🌎 Интернет
- 📱 Устройство
- 📬 Входящие данные

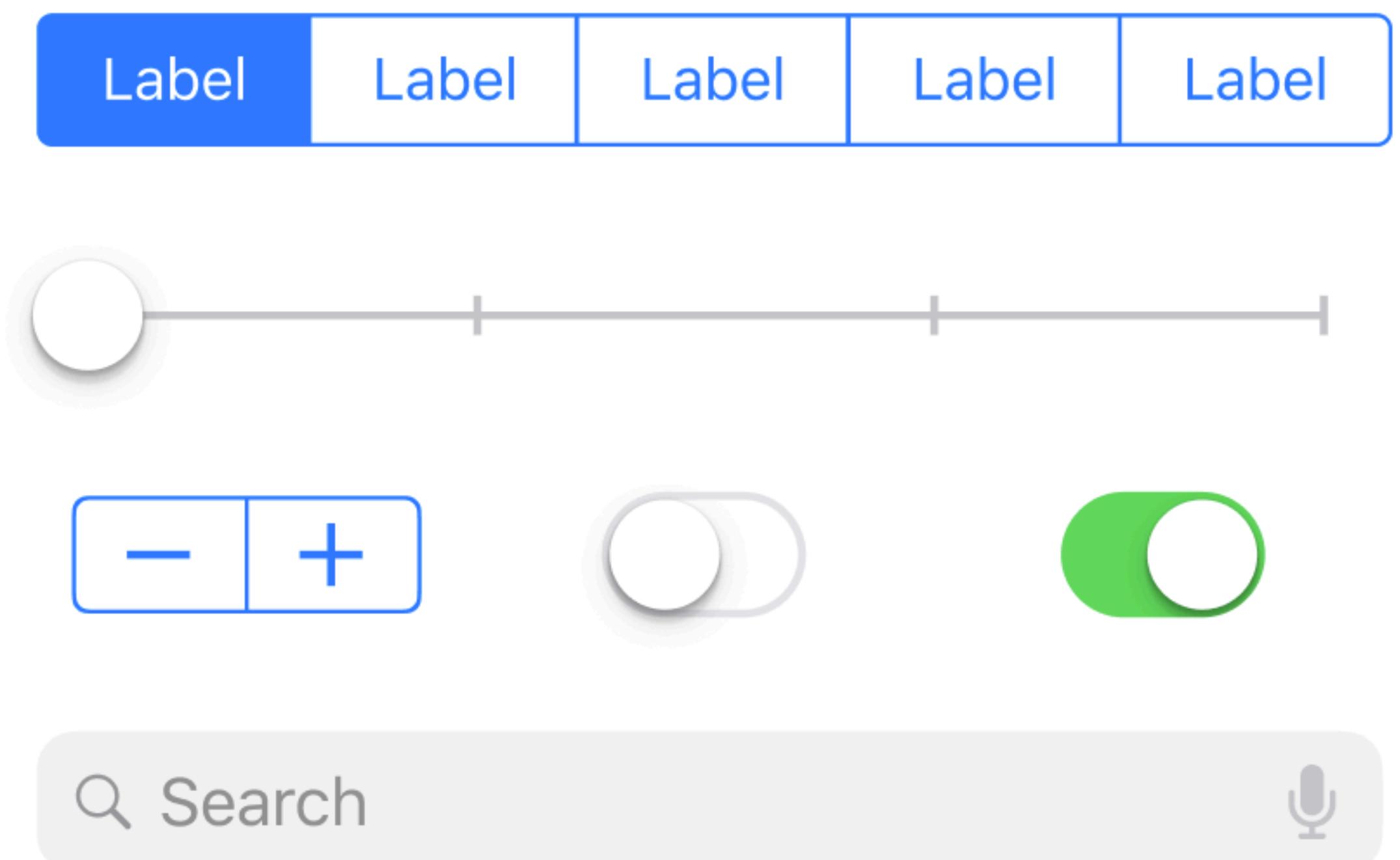


# От пользователя

- UI элементы  
*UITextField, UISwitch, ...*
- Буфер обмена  
*UIPasteboard*
- Камера, микрофон  
*AVFoundation*

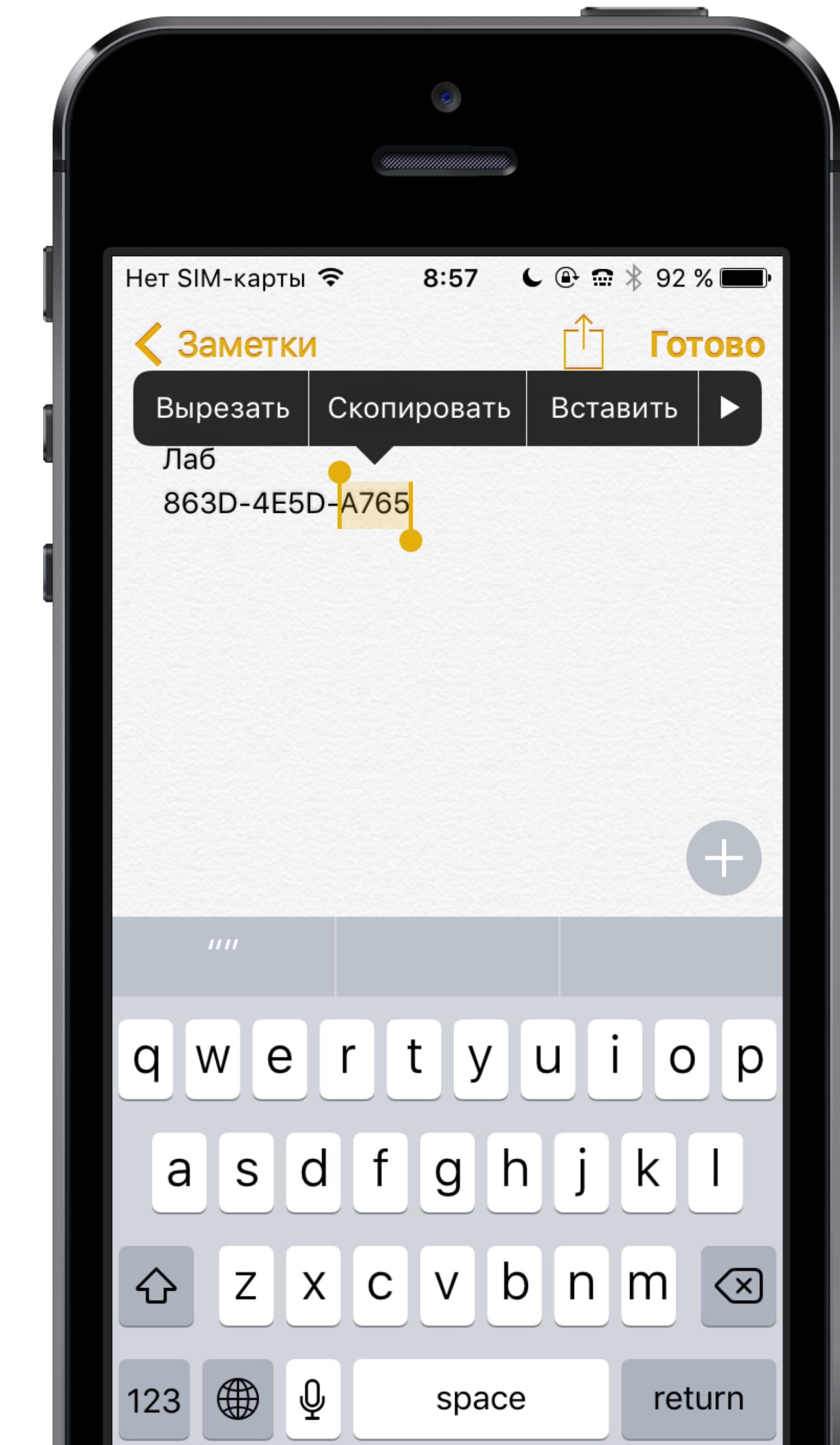
# От пользователя

- **UI элементы**  
*UITextField, UISwitch, ...*
- Буфер обмена  
*UIPasteboard*
- Камера, микрофон  
*AVFoundation*



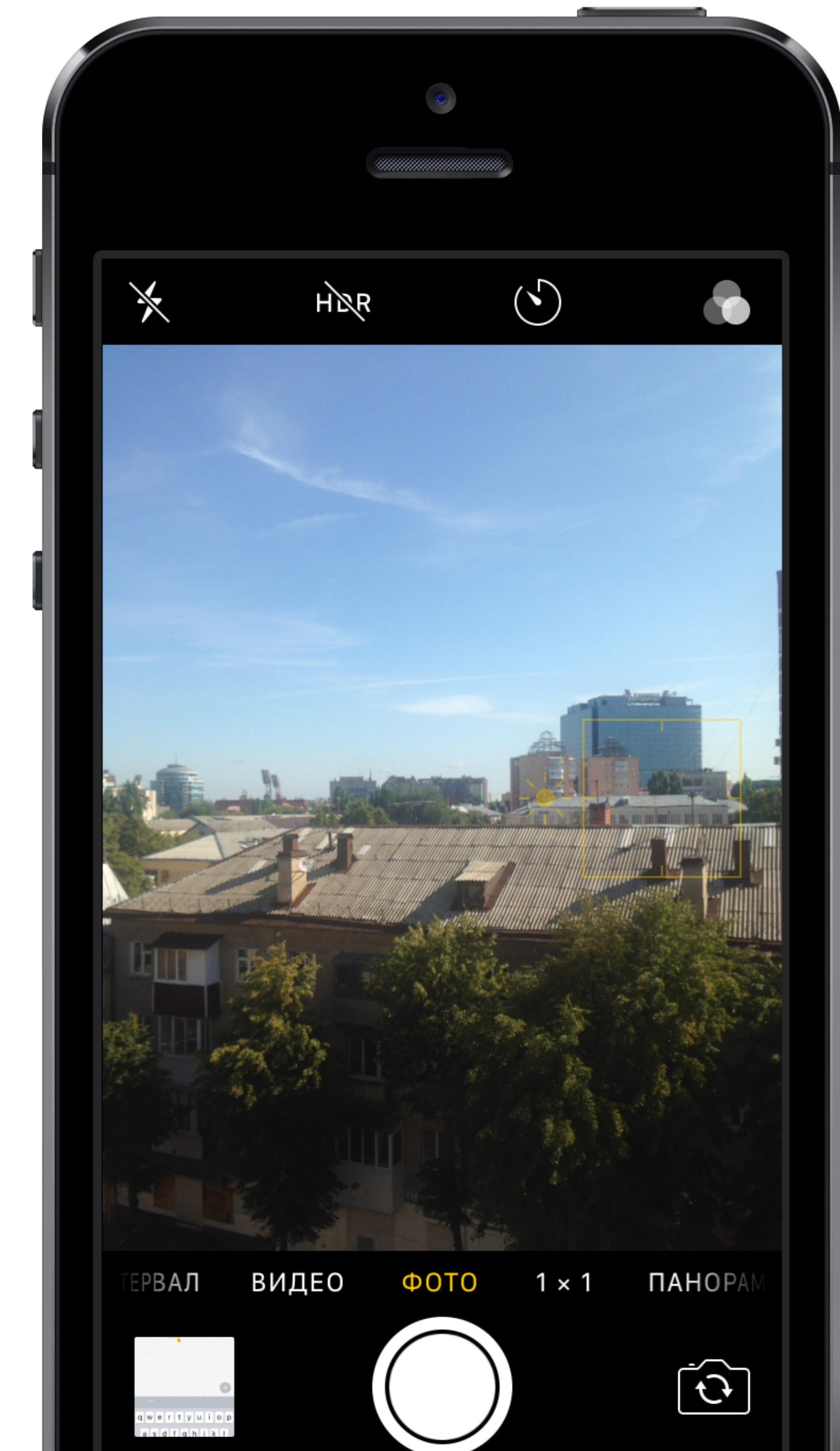
# От пользователя

- UI элементы  
*UITextField, UISwitch, ...*
- **Буфер обмена**  
*UIPasteboard*
- Камера, микрофон  
*AVFoundation*



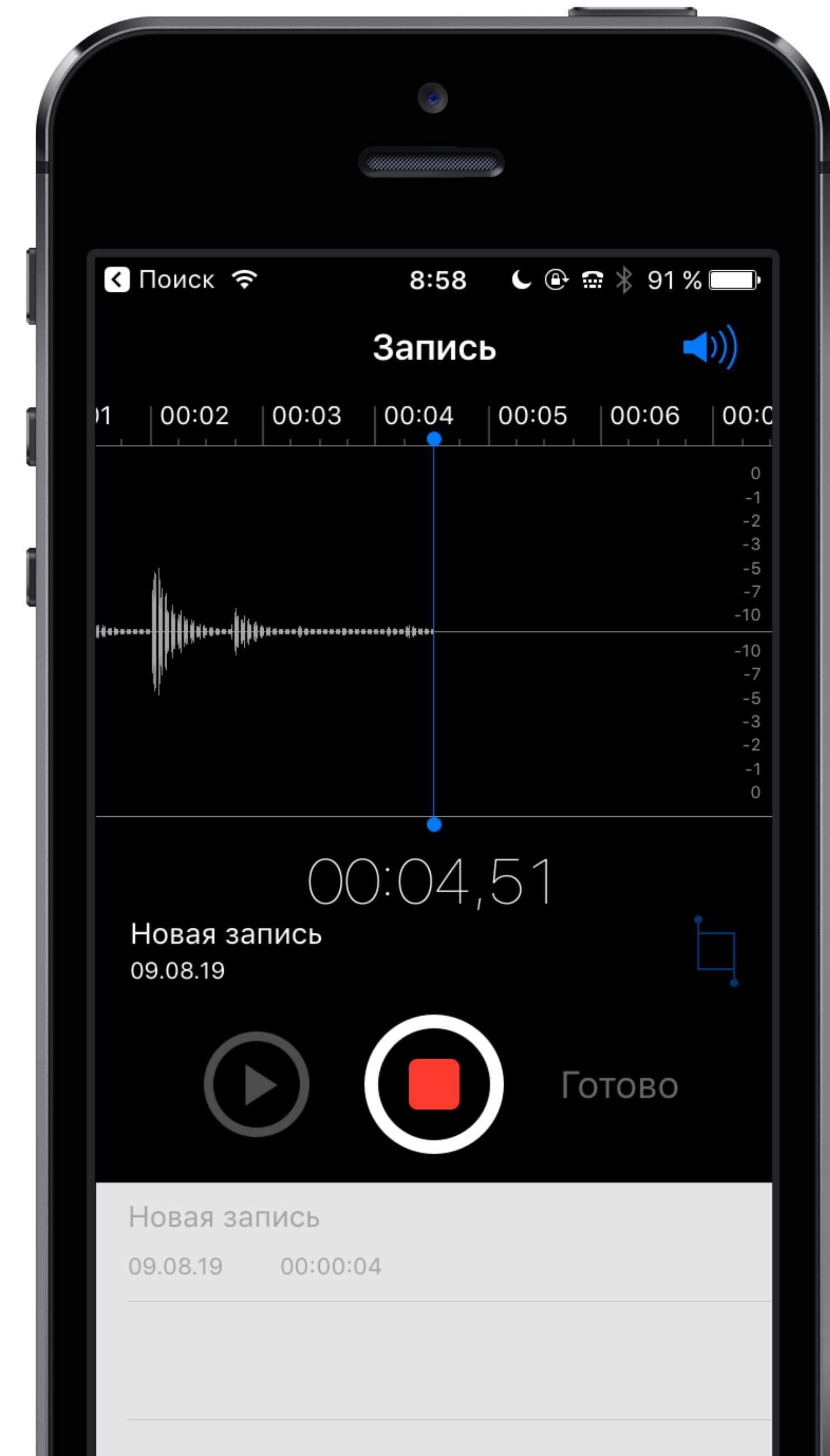
# От пользователя

- UI элементы  
*UITextField, UISwitch, ...*
- Буфер обмена  
*UIPasteboard*
- **Камера, микрофон**  
*AVFoundation*



# От пользователя

- UI элементы  
*UITextField, UISwitch, ...*
- Буфер обмена  
*UIPasteboard*
- Камера, **микрофон**  
*AVFoundation*



# Из интернета

- Типы API
- Нативная работа с сетью в iOS
- Сторонние библиотеки
- Инструменты

# Типы API

- REST
- SOAP
- WebSocket
- gRPC
- GraphQL
- ...

# Типы API

- REST

- SOAP

- **WebSocket**

- gRPC

- GraphQL

- ...

# **HTTP URL**

<https://api.unsplash.com/photos/random>

# **HTTP URL**

`https://api.unsplash.com/photos/random`



Protocol

# **HTTP URL**

`https://api.unsplash.com/photos/random`



Адрес сервера

# **HTTP URL**

`https://151.101.53.181/photos/random`



Адрес сервера

# HTTP URL

`https://api.unsplash.com/photos/random`



Единица информации

# HTTP URL

`https://api.unsplash.com/photos/random`



Конкретизация единицы

# **HTTP URL**

`https://api.unsplash.com/photos/`



Конкретизация единицы

# HTTP URL

`https://api.unsplash.com/photos/123842`



Конкретизация единицы

# HTTP URL

`https://api.unsplash.com/photos/1?client_id=21`



Дополнительные параметры

# Нативная работа с сетью в iOS



# Сторонние библиотеки

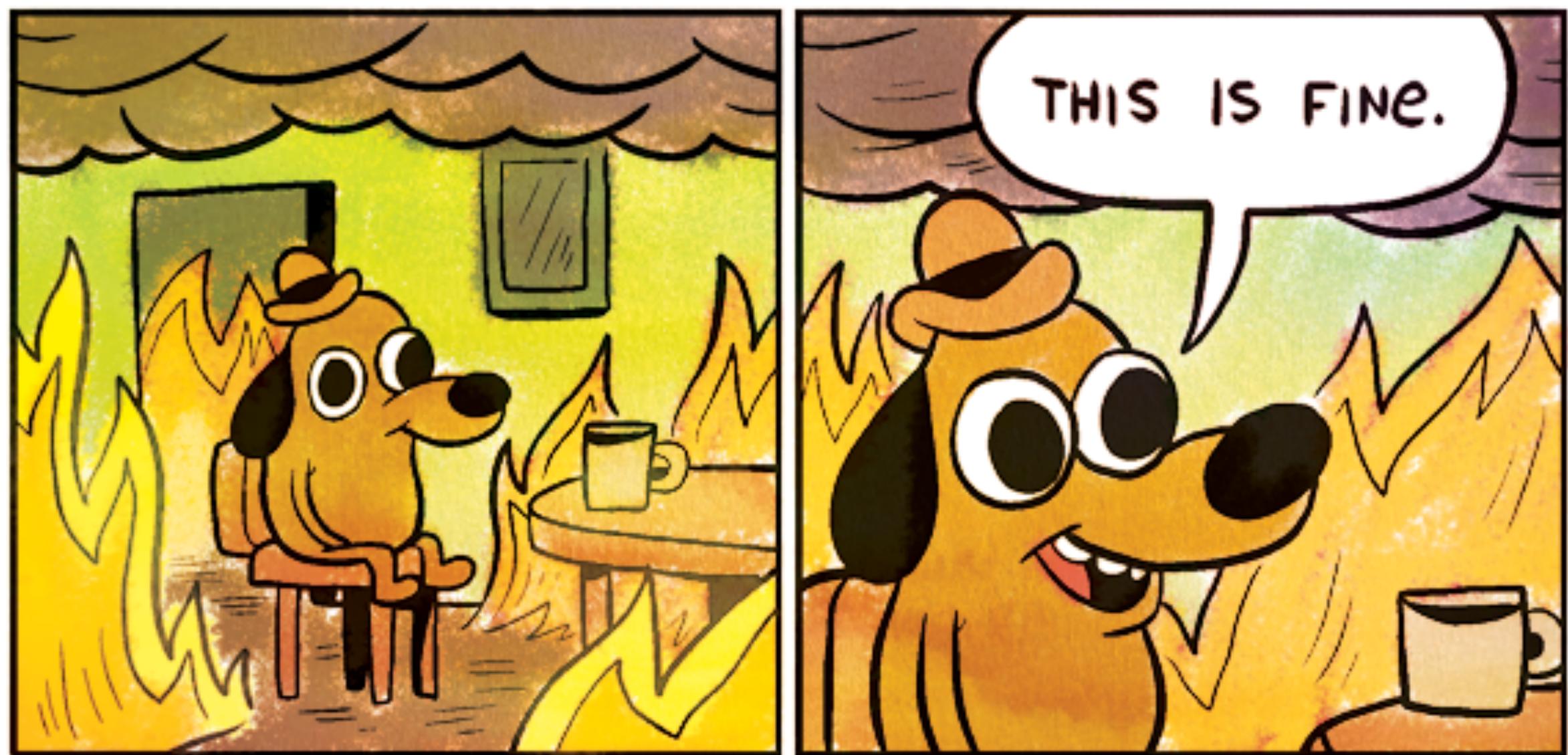
- Для работы с сетью
  - Alamofire
  - Moya
  - **NodeKit (by Surf)**
- Для загрузки изображений
  - Alamofire+Image
  - Nuke

# **Инструменты**

- Postman
- Charles
- Paw

# Из интернета

- Типы API
- Нативная работа с сетью в iOS
- Сторонние библиотеки
- Инструменты



# Из устройства

- События системы и системная информация

*CoreWLAN, Core Telephony, ...*

- Системные фреймворки

*AddressBook, CallKit, MapKit, ...*

# Входящие данные

- Push Notifications  
*Local/Remote*
- DeepLinks  
*UniversalLinking, DynamicLinks, ...*
- Handoff
- «Открыть в ...»  
*UTI в приложении*

# Входящие данные

- **Push Notifications**

*Local/Remote*

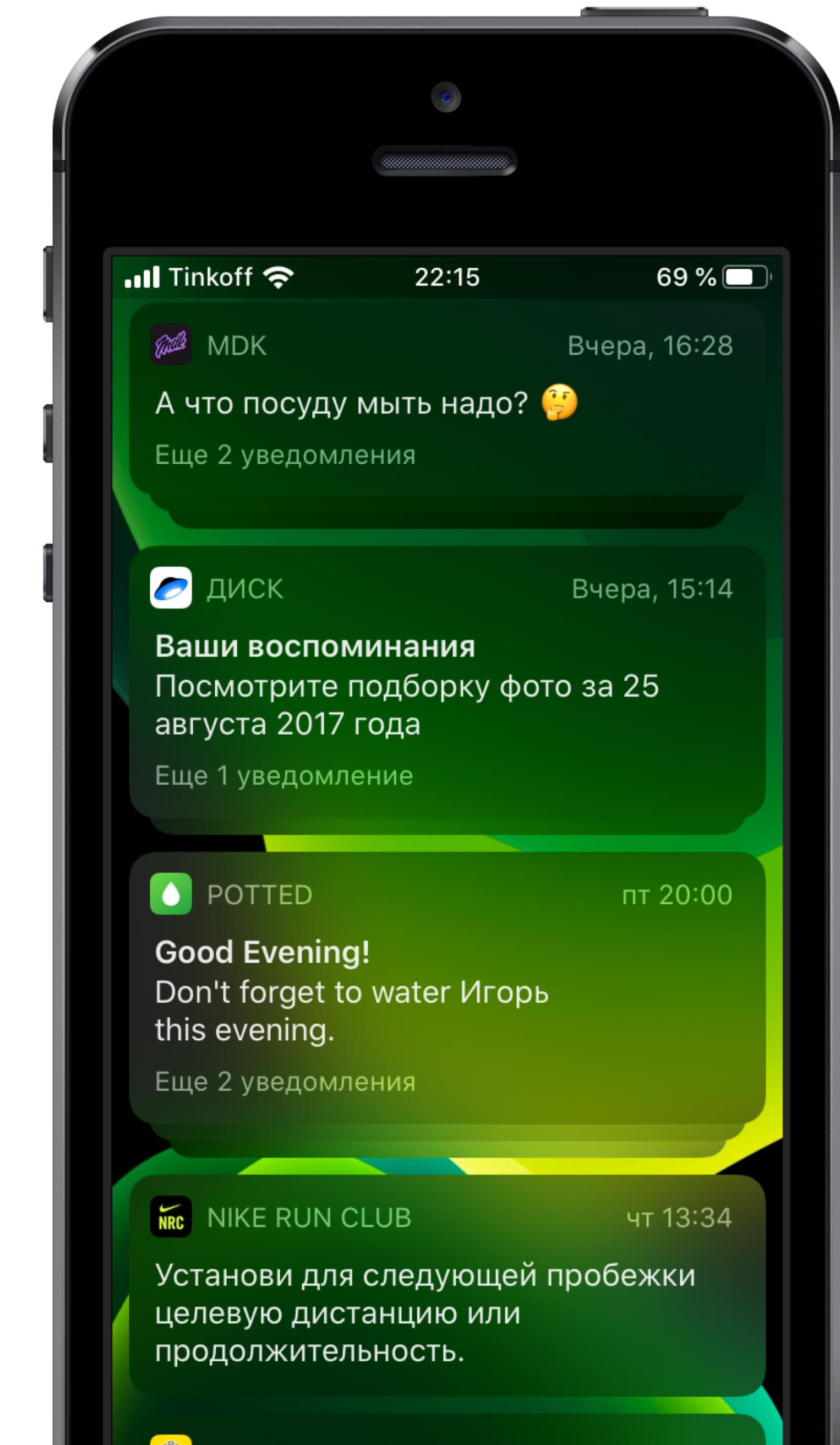
- **DeepLinks**

*UniversalLinking, DynamicLinks, ...*

- **Handoff**

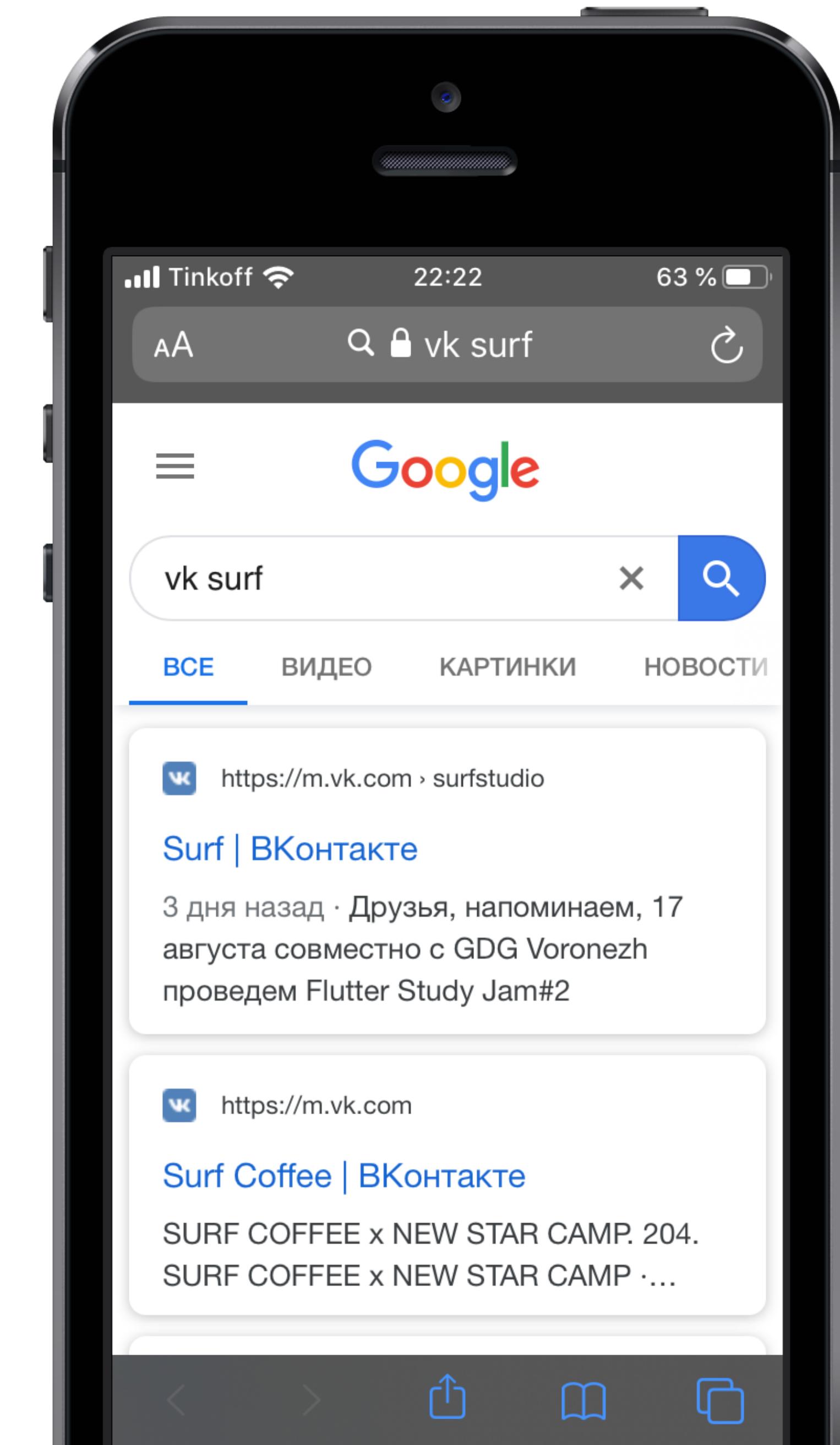
- **«Открыть в ...»**

*UTI в приложении*



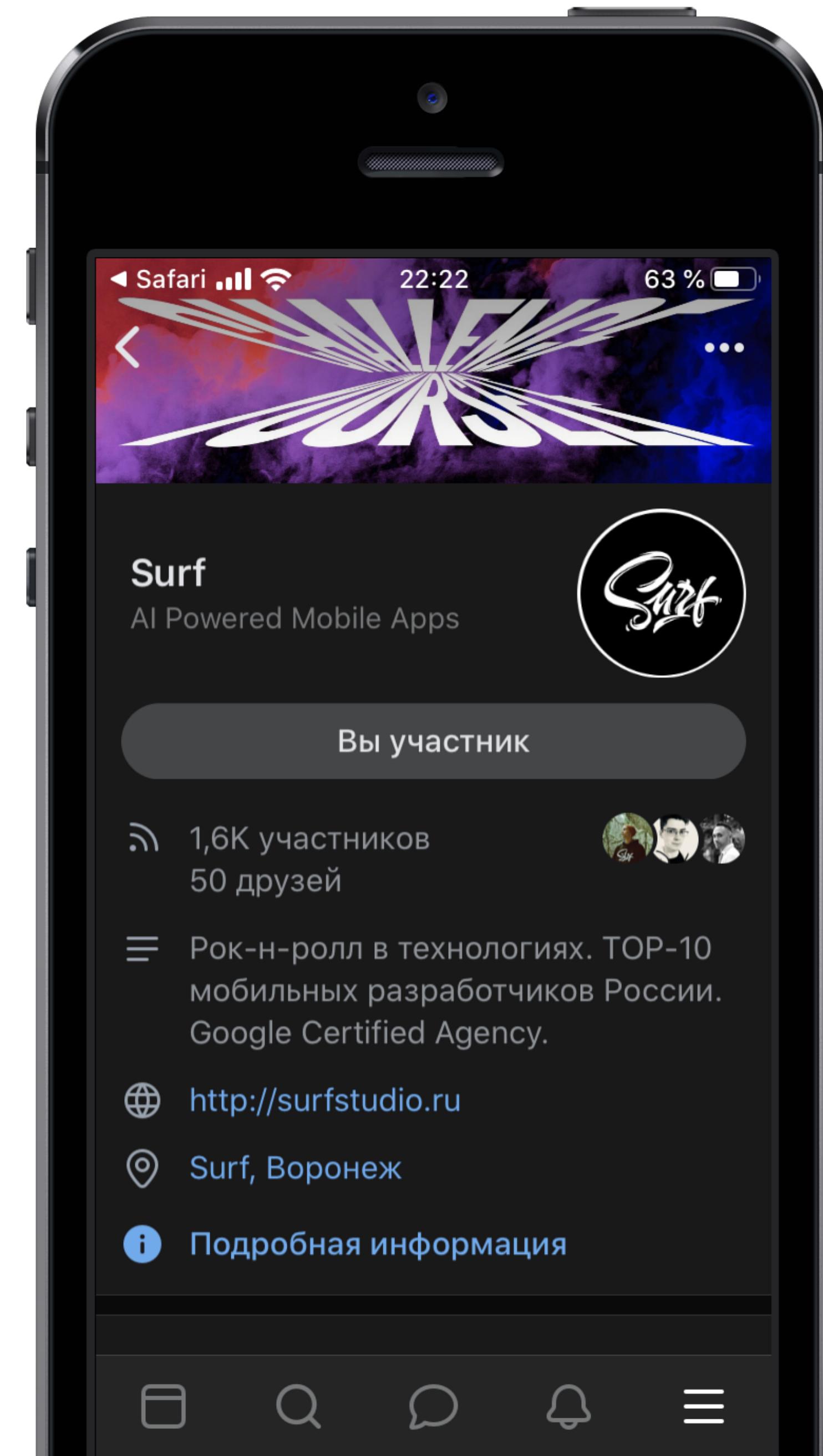
# Входящие данные

- Push Notifications  
*Local/Remote*
- **DeepLinks**  
*UniversalLinking, DynamicLinks, ...*
- Handoff
- «Открыть в ...»  
*UTI в приложении*



# Входящие данные

- Push Notifications  
*Local/Remote*
- **DeepLinks**  
*UniversalLinking, DynamicLinks, ...*
- Handoff
- «Открыть в ...»  
*UTI в приложении*



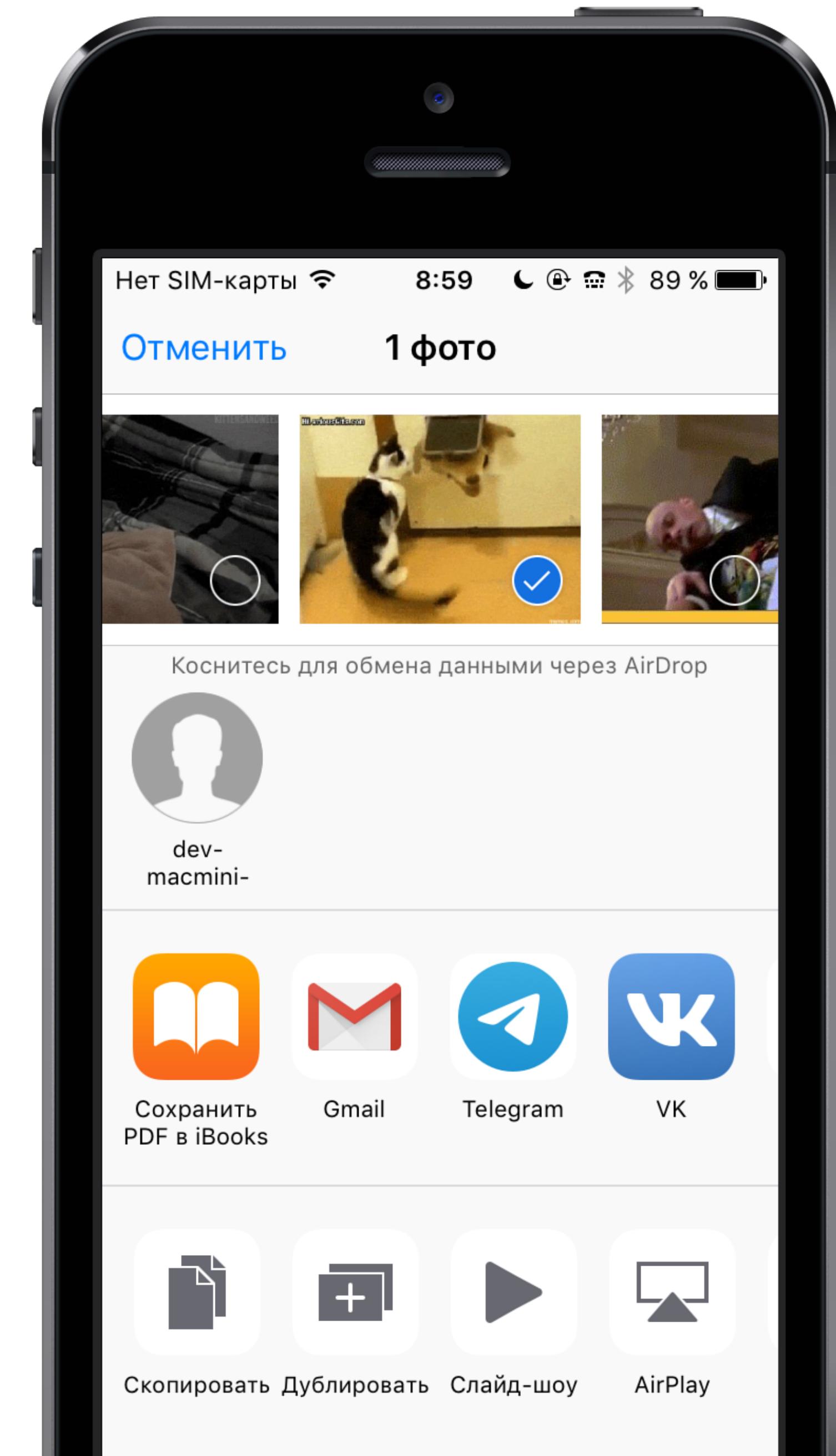
# Входящие данные

- Push Notifications  
*Local/Remote*
- DeepLinks  
*UniversalLinking, DynamicLinks, ...*
- Handoff
- «Открыть в ...»  
*UTI в приложении*



# Входящие данные

- Push Notifications  
*Local/Remote*
- DeepLinks  
*UniversalLinking, DynamicLinks, ...*
- Handoff
- «Открыть в ...»  
*UTI в приложении*



**Классификация данных**

**Получение данных**

**Отправка данных**

**Обработка данных**

**Классификация данных**

**Получение данных**

**Отправка данных**

**Обработка данных**

# Отправка данных

- На сервер
- В буфер обмена
- Share
- Обмен данных между приложениями

**Классификация данных**

**Получение данных**

**Отправка данных**

**Обработка данных**

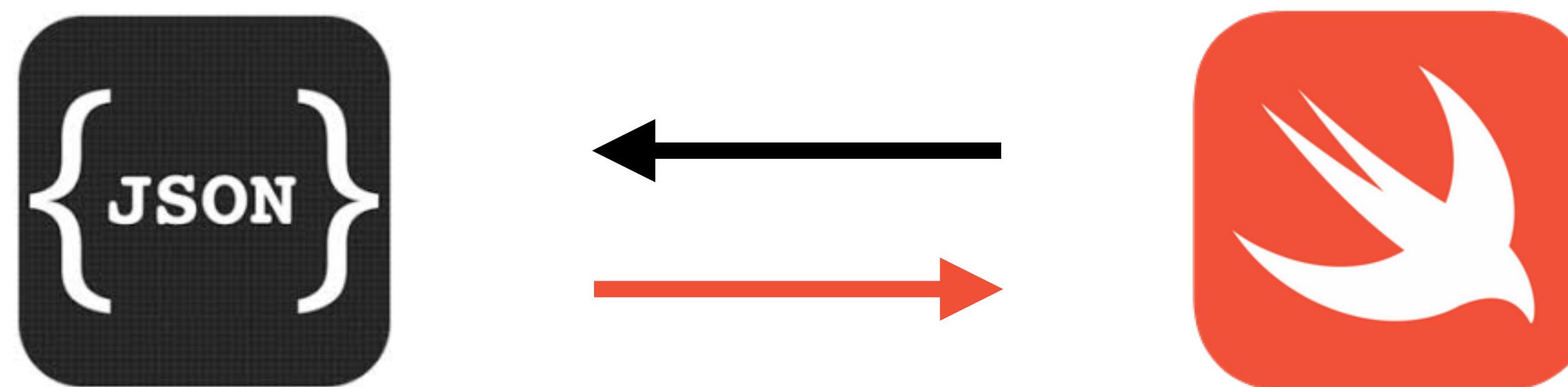
**Классификация данных**

**Получение данных**

**Отправка данных**

**Обработка данных**

# Маппинг данных



# JSON объект фотографии

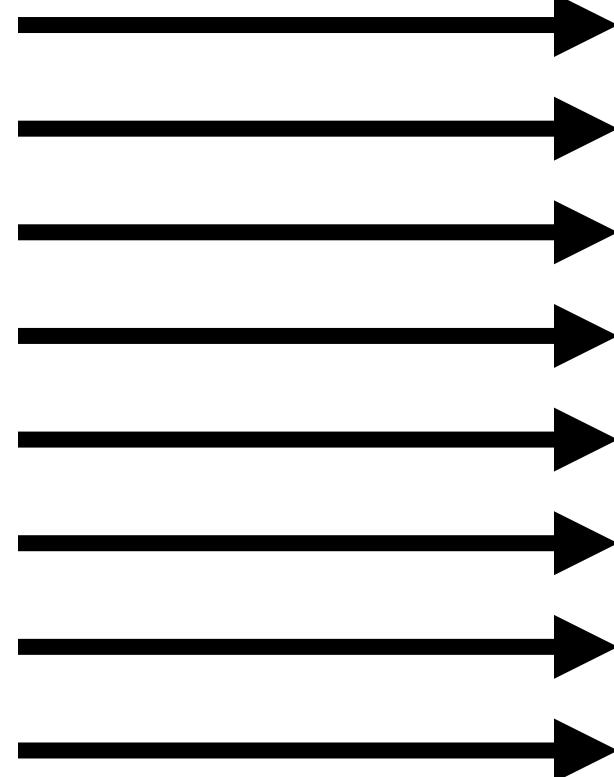
```
{
 "id": "M65Hq1G2inc",
 "width": 3456,
 "height": 4608,
 "description": "hasselblad",
 "urls": {...},
 "user": {...},
 "exif": {...},
 "views": 189525
}
```

# Структура фотографии в Swift

```
struct Photo {
 let id: String
 let width: Int
 let height: Int
 let urls:Urls
 let user:User
 let exif:Exif
 let description:String
 let views:Int
}
```

# Сопоставление полей

```
{
 "id": "M65HqlG2inc",
 "width": 3456,
 "height": 4608,
 "description": "hasselblad",
 "urls": {...},
 "user": {...},
 "exif": {...},
 "views": 189525
}
```



```
struct Photo {
 let id: String
 let width: Int
 let height: Int
 let urls:Urls
 let user:User
 let exif:Exif
 let description:String
 let views:Int
}
```

# Codable = Encodable & Decodable

```
struct Photo: Codable {
 let id: String
 let width: Int
 let height: Int
 let urls:Urls
 let user:User
 let exif:Exif
 let description:String
 let views:Int
}
```

# **Codable = Encodable & Decodable**

```
structUrls: Codable {
}
```

```
structUser: Codable {
}
```

```
structExif: Codable {
}
```

**Классификация данных**

**Получение данных**

**Отправка данных**

**Обработка данных**

# **Классификация данных**

## **Получение данных**

## **Отправка данных**

## **Обработка данных**

# Материалы

<http://bit.ly/iossurf19lecture6>

