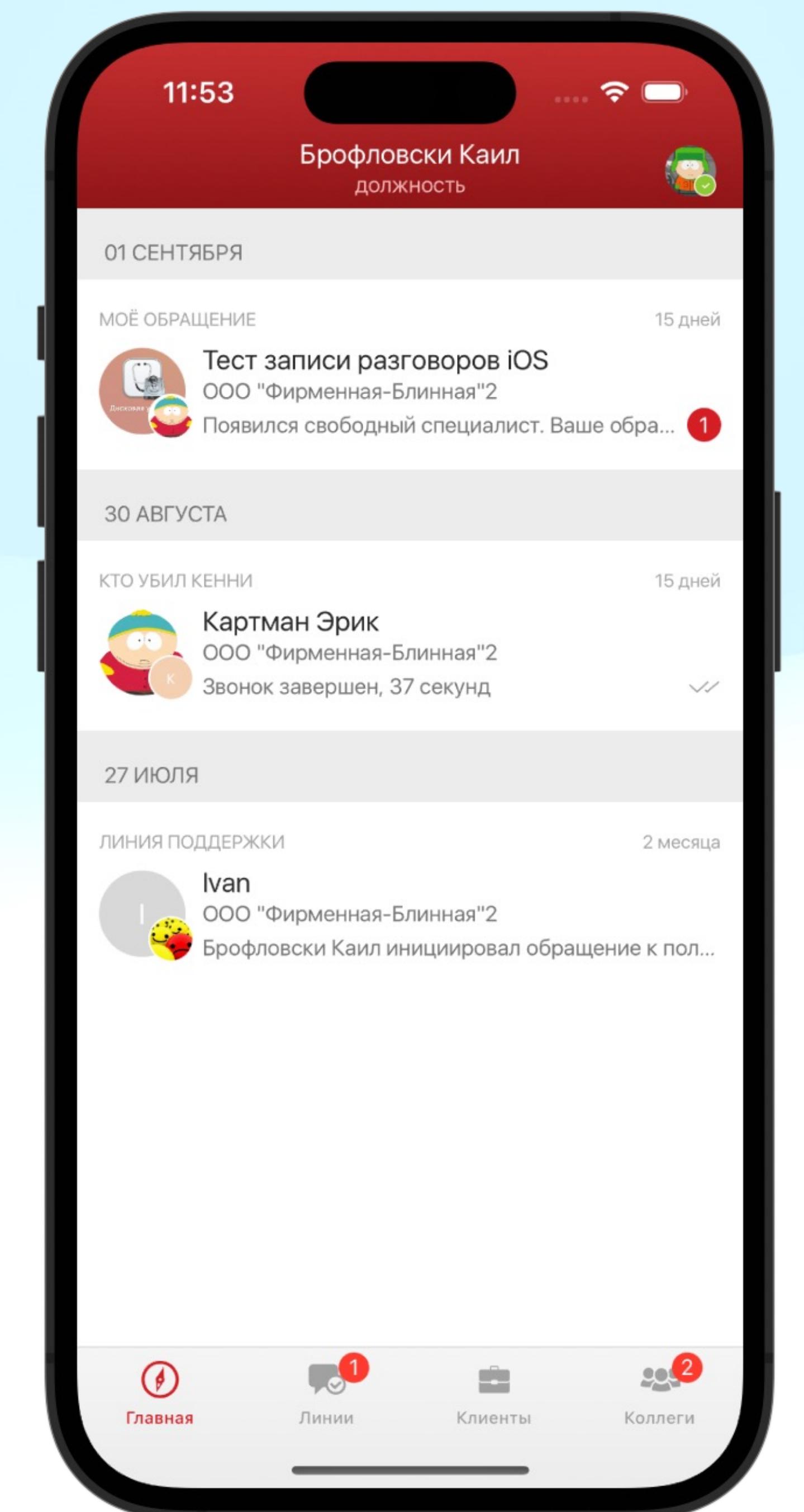


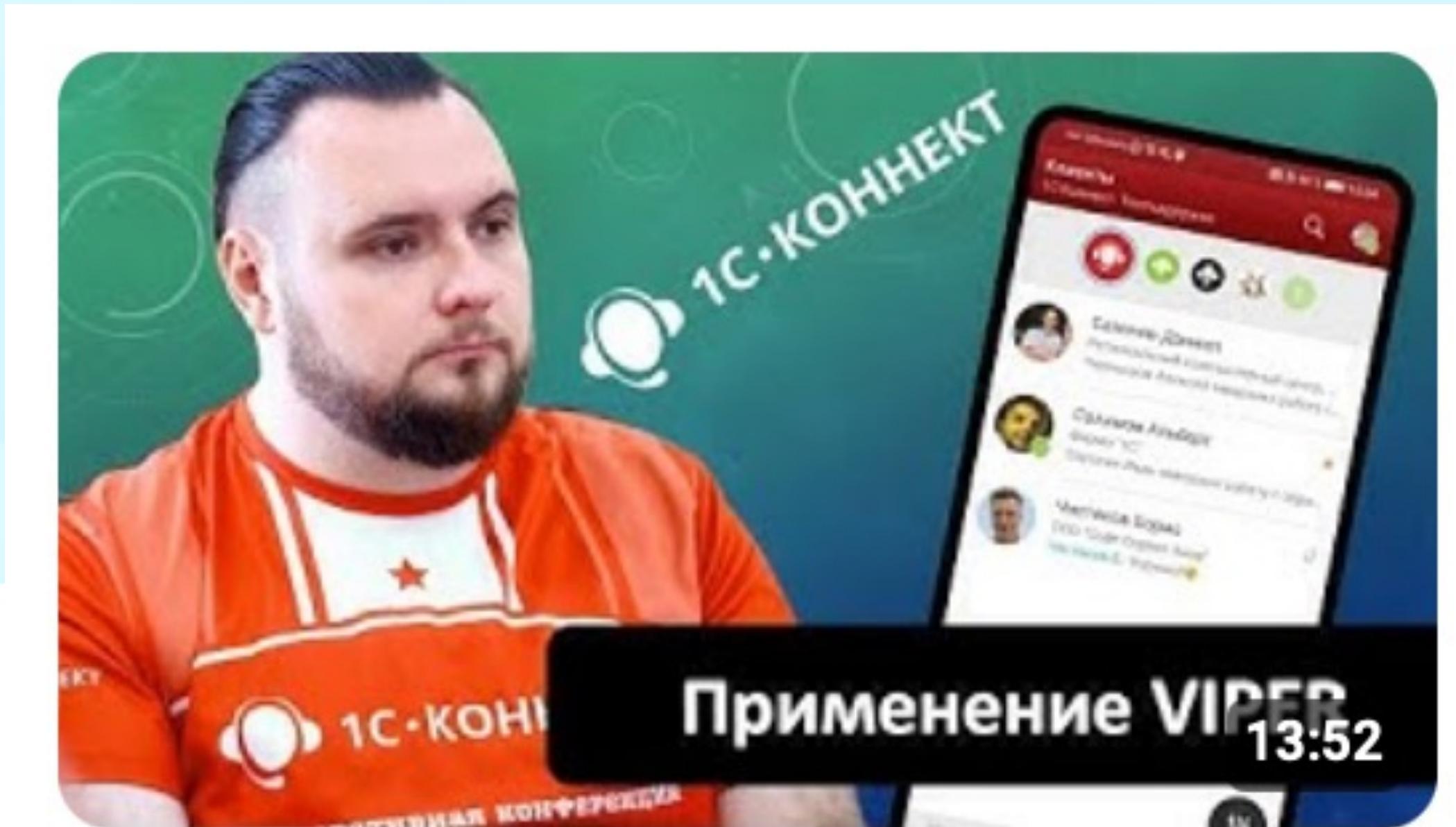
Как мы (не)выпустили CarPlay



Сергей Долгих 13 сентября 2024 г.
iOS разработчик, 1C-Коннект



Имя	Тип	Размер
> _CodeSignature	Папка	--
> en.lproj	Папка	--
> Frameworks	Папка	--
Plugins	Папка	--
IntentsExtension.appex	Расширение программ и системы	11,6 МБ
NotificationContent.appex	Расширение программ и системы	11,3 МБ
NotificationService.appex	Расширение программ и системы	12 МБ
ShareExtension.appex	Расширение программ и системы	11,3 МБ
> ru.lproj	Папка	--
> SC_Info	Папка	--
outgoing_call.mp3	Аудио MP3	242 КБ
Assets.car	Документ	8,5 МБ
AuthorizationStateView.nib	Документ	6 КБ
ChatDateHeader.nib	Документ	4 КБ
ChatInputView.nib	Документ	9 КБ
ChatPreloaderView.nib	Документ	1 КБ
ChatQuoteView.nib	Документ	5 КБ



Опыт разработки iOS. Применение VIPER. :

<https://youtu.be/-RqNPshitEQ?si=tvyA5NDtXleOmsL>



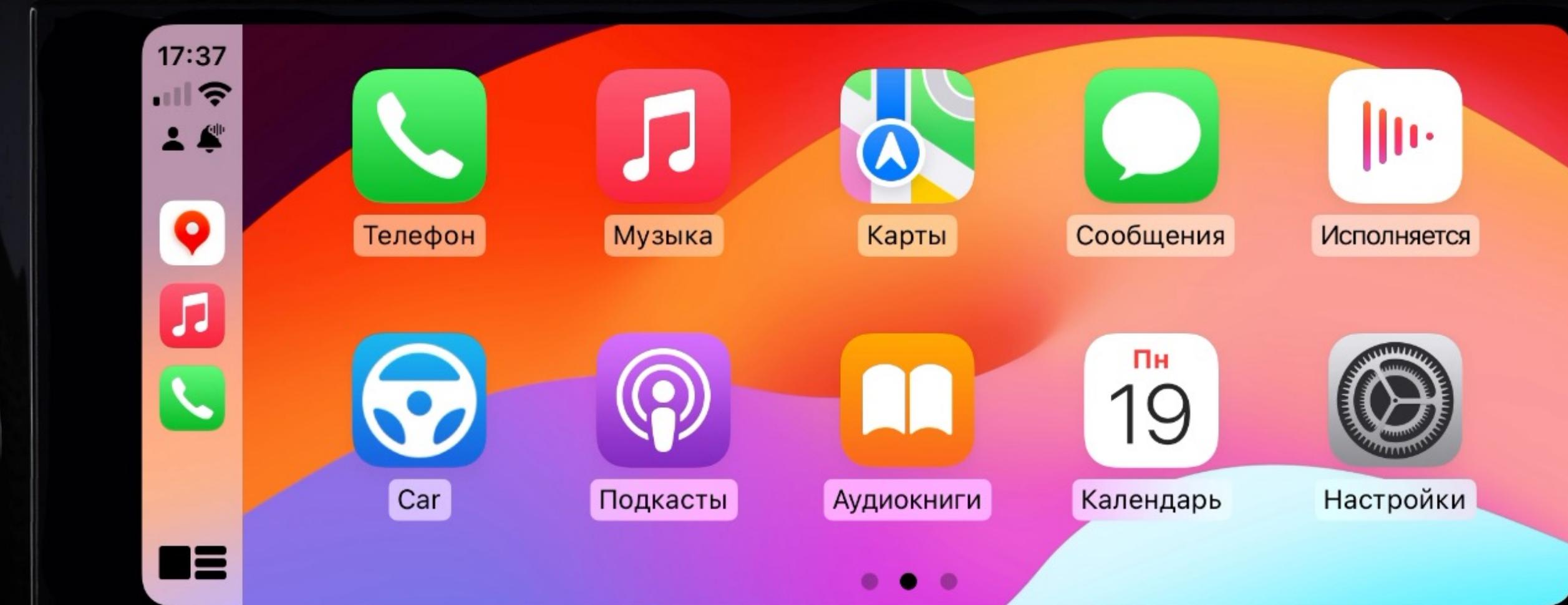


Ни слова больше



Добавляем CarPlay

Что такое CarPlay?



Изучаем документацию

developer.apple.com

Apple Developer News Discover Design Develop Distribute Support Account

Documentation Language: Swift API changes: None

CarPlay / Requesting CarPlay Entitlements

Article

Requesting CarPlay Entitlements

Configure your CarPlay-enabled app with the entitlements it requires.

Overview

To integrate with CarPlay, you must request the appropriate entitlement for your app's category at [CarPlay Contact Us](#) and agree to the CarPlay Entitlement Addendum. Apple reviews each application using predefined criteria. If your request meets the criteria, Apple adds the entitlement to your developer account using managed capabilities. For more information, see [Provisioning with managed capabilities](#).

After you receive the entitlement, you need to configure your Xcode project to use it, which involves several steps. You create or update an App ID, generate a provisioning profile, and add an Entitlements.plist file to your target. Your project's code signing settings also require minor changes.

Developer Information

Name

Sergey Dolgikh

Email

ageres7@gmail.com

Organization

BukhgalterFon Servis, OOO - 7B4P54567H ▾

Additional Details

App Type

Communication (messaging and calling) ▾

Tell us about your app.

Communication apps that provide text messaging features must support all 3 of the following SiriKit intents:

- Send a message ([INSendMessageIntent](#))
- Request a list of messages ([INSearchForMessagesIntent](#))
- Modify the attributes of a message ([INSetMessageAttributeIntent](#))

Communication apps that provide VoIP calling features must support CallKit, and all of the following SiriKit intents:

- Start a call ([INStartCallIntent](#))
- Start an audio-only call ([INStartAudioCallIntent](#)) required for apps that support iOS 14 and earlier
- Request a list of calls ([INSearchCallHistoryIntent](#)) required for apps that support iOS 14 and earlier

- **INSendMessageIntent**
- **INSearchForMessagesIntent**
- **INSetMessageAttributeIntent**
- **INStartCallIntent**

Дорабатываем IntentsExtension

И в итоге IntensExtension умеет:

- Выполнять обновление данных для поиска по актуальным контактам и группам

И в итоге IntensExtension умеет:

- Выполнять обновление данных для поиска по актуальным контактам и группам
- Искать контакт по имени и просить пользователя выбрать нужного

И в итоге IntensExtension умеет:

- Выполнять обновление данных для поиска по актуальным контактам и группам
- Искать контакт по имени и просить пользователя выбрать нужного
- Проверять сообщение, которое будет отправляться

И в итоге IntensExtension умеет:

- Выполнять обновление данных для поиска по актуальным контактам и группам
- Искать контакт по имени и просить пользователя выбрать нужного
- Проверять сообщение, которое будет отправляться
- Отправлять сообщения

И в итоге IntensExtension умеет:

- Выполнять обновление данных для поиска по актуальным контактам и группам
- Искать контакт по имени и просить пользователя выбрать нужного
- Проверять сообщение, которое будет отправляться
- Отправлять сообщения
- Зачитывать новые сообщения из нужного чата

И в итоге IntensExtension умеет:

- Выполнять обновление данных для поиска по актуальным контактам и группам
- Искать контакт по имени и просить пользователя выбрать нужного
- Проверять сообщение, которое будет отправляться
- Отправлять сообщения
- Зачитывать новые сообщения из нужного чата
- Определять, кому хочет позвонить пользователь и перенаправлять намерение в основное приложение

Передача в систему имен коллег и название групп

```
1 func updateContact(from contactModels: [ContactModelProtocol]) {
2     let speakableStrings: [INSpeakableString] = contactModels
3         .sorted(by: Compares.contactByMessageTime)
4         .compactMap {
5             guard
6                 let identifier = $0.agent.identifier,
7                 let name = $0.agent.fullName.fullFIO
8             else { return nil }
9
10            let id = IntentContactID(chatType: .p2p, chatId: identifier)
11
12            return INSpeakableString(
13                vocabularyIdentifier: id.rawValue,
14                spokenPhrase: name,
15                pronunciationHint: nil
16            )
17        }
18        let contacts = NSOrderedSet(array: speakableStrings)
19
20        INVocabulary.shared().setVocabulary(contacts, of: .contactName)
21 }
```

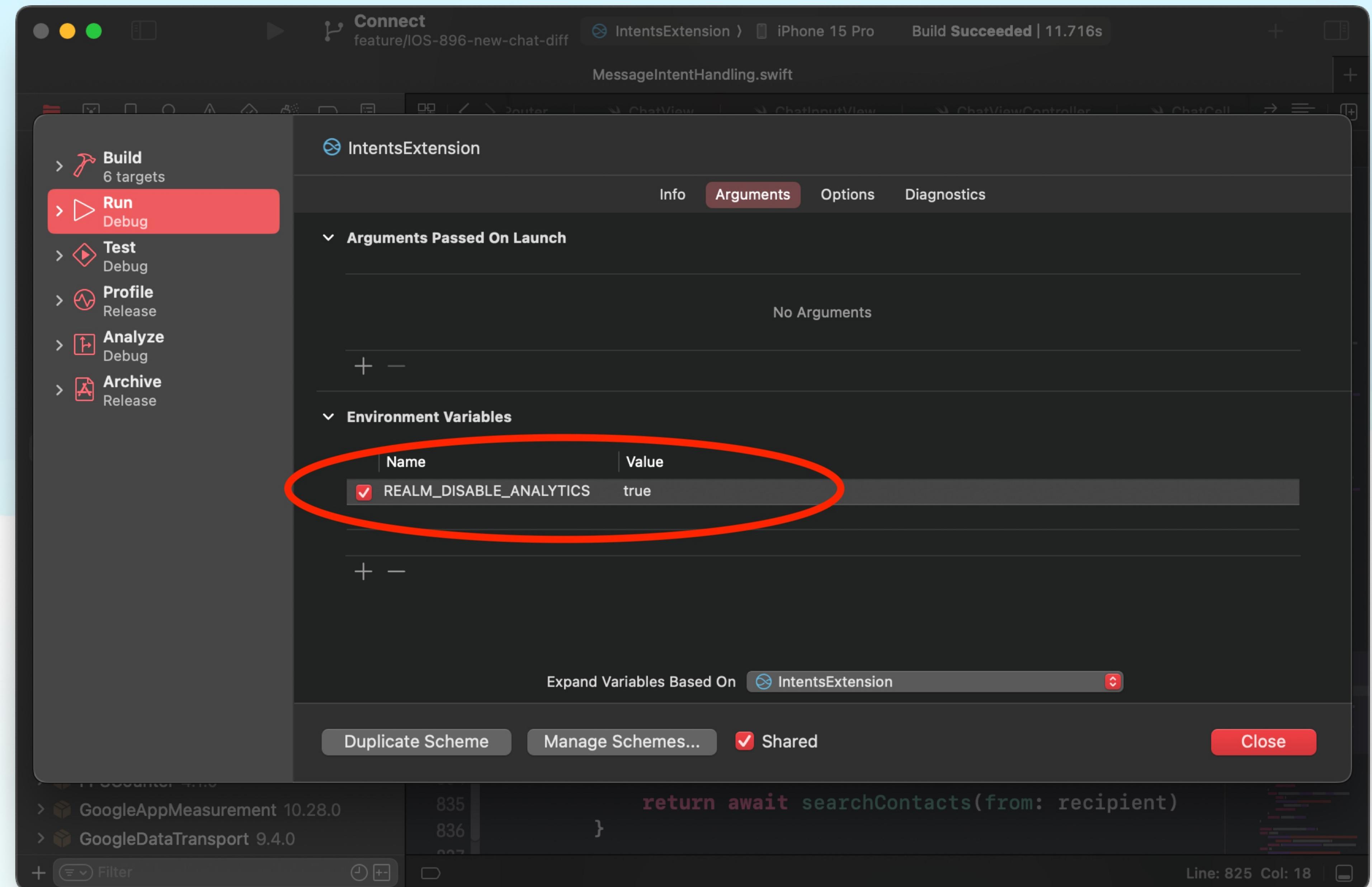
Первый запуск на устройстве...

30 МБ оперативной памяти...

**Из которых 15 МБ уже заняты
самой Siri**

Оптимизация кода

Отключение отправки аналитики базы данных Realm



Создание индекса для поиска контактов

```
1  class BaseChatInfoRealmModel: Object {  
2      @objc dynamic var identifier: String?  
3      @objc dynamic var name: String?  
4      @objc dynamic var phoneNumber: String?  
5      @objc dynamic var activeTime: Date?  
6      @objc dynamic var type: ChatTypeRealm = .unknown  
7  
8      override static func primaryKey() -> String? {  
9          return "identifier"  
10     }  
11 }
```

Отключение кеширования в URL Session

```
1 let urlSession: URLSession = {
2     let session: URLSession
3 #if TARGET_MAIN_APP
4     session = URLSession.shared
5 #else
6     session = URLSession(configuration: .ephemeral)
7     session.configuration.urlCache?.memoryCapacity = 128
8 #endif
9     return session
10 }()
```

Отключение кэширования в память картинок в Kingfisher

```
1 func downloadImage(url: URL) async throws -> Data {
2     try await withCheckedThrowingContinuation { (continuation: CheckedContinuation<Data, Error>) in
3         let downloader = ImageDownloader.default
4 #if TARGET_INTENTS
5             downloader.sessionConfiguration.urlCache?.memoryCapacity = 128
6 #endif
7         let cache = ImageCache.default
8
9 #if !TARGET_MAIN_APP
10            cache.memoryStorage.config.totalCostLimit = 0
11 #endif
12            cache.diskStorage.config.expiration = .days(30)
13
14 if let imageData = try? cache.diskStorage.value(forKey: url.absoluteString) {
15
16     Logger.info("Complete get image from cache \(url)")
17     continuation.resume(returning: imageData)
18
19 } else {
20     downloader.downloadImage(with: url) { downloadResult in
21         switch downloadResult {
22
23             case .success(let value):
24                 Logger.info("Complete download image \(url)")
25                 cache.storeToDisk(value.originalData, forKey: url.absoluteString)
```

**На этом моменте казалось,
что финал близок**

**На тестовом телефоне все
работает...**

Ho...

**На личных телефонах заставить
Siri отправить сообщение через
1С-Коннект практически
невозможно**

Почему?

**Кроме имен нужно передавать
в систему Intent**

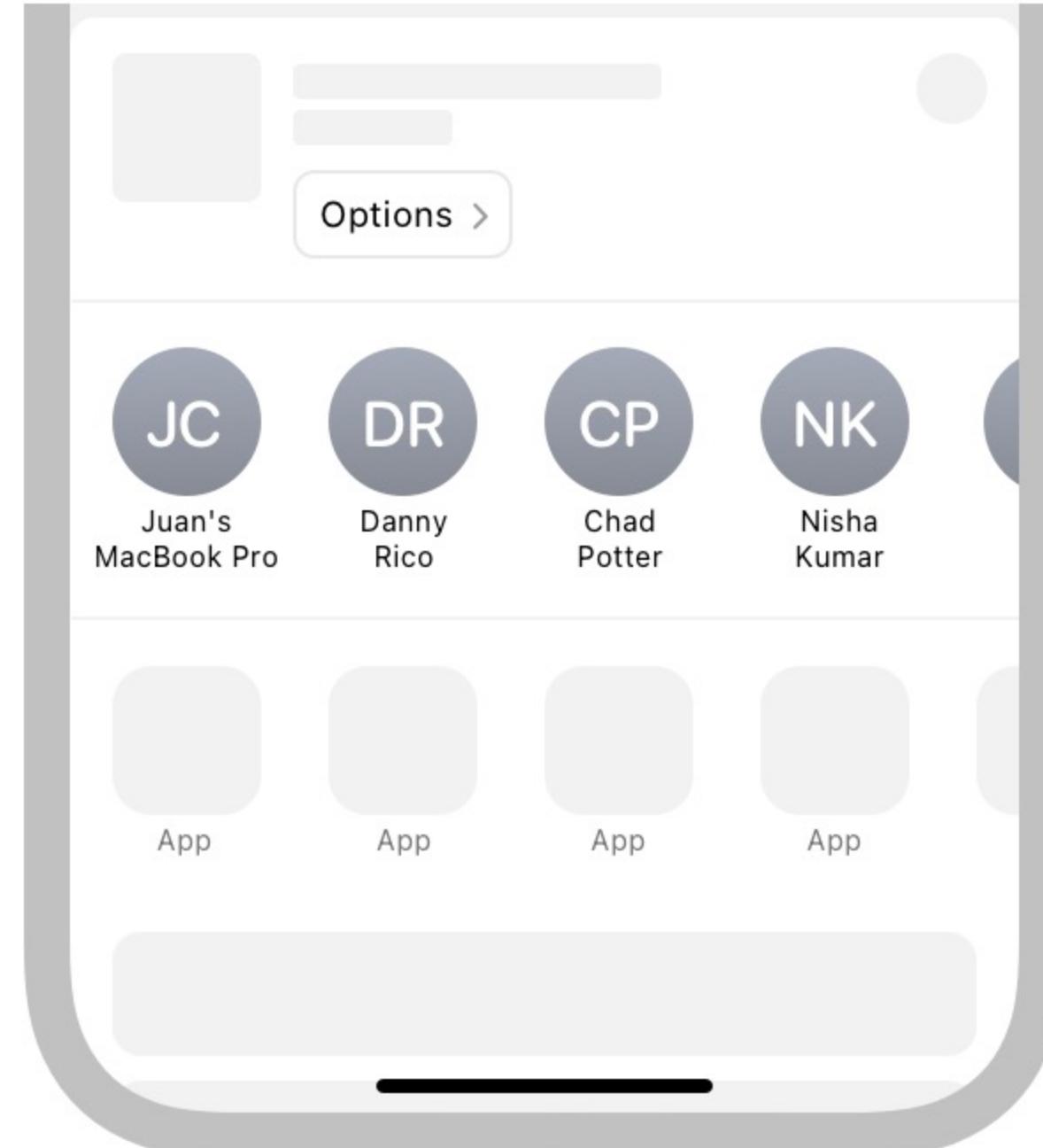
```
let id = IntentContactID(chatType: chatType, chatId: chatId)
let messageIntent = INSendMessageIntent(
    recipients: recipients,
    outgoingMessageType: .outgoingMessageText,
    content: message,
    speakableGroupName: groupName,
    conversationIdentifier: id.rawValue,
    serviceName: nil,
    sender: sender,
    attachments: nil
)

if let conferenceImage {
    messageIntent.setImage(conferenceImage, forParameterNamed: \.speakableGroupName)
}
await donateInteraction(intent: messageIntent, direction: .outgoing, id: id)
```

```
static func suggestCall(  
    with agent: AgentModel,  
    direction: INInteractionDirection  
) async {  
    guard let agentId = agent.identifier else { return }  
  
    let person = await createINPerson(agentId: agentId, agent: agent)  
  
    let intent = INStartCallIntent(  
        callRecordFilter: nil,  
        callRecordToCallBack: nil,  
        audioRoute: .speakerphoneAudioRoute,  
        destinationType: .normal,  
        contacts: [person],  
        callCapability: .audioCall  
    )  
    let id = IntentContactID(chatType: .p2p, chatId: agentId)  
  
    await donateInteraction(intent: intent, direction: direction, id: id)  
}
```

developer.apple.com

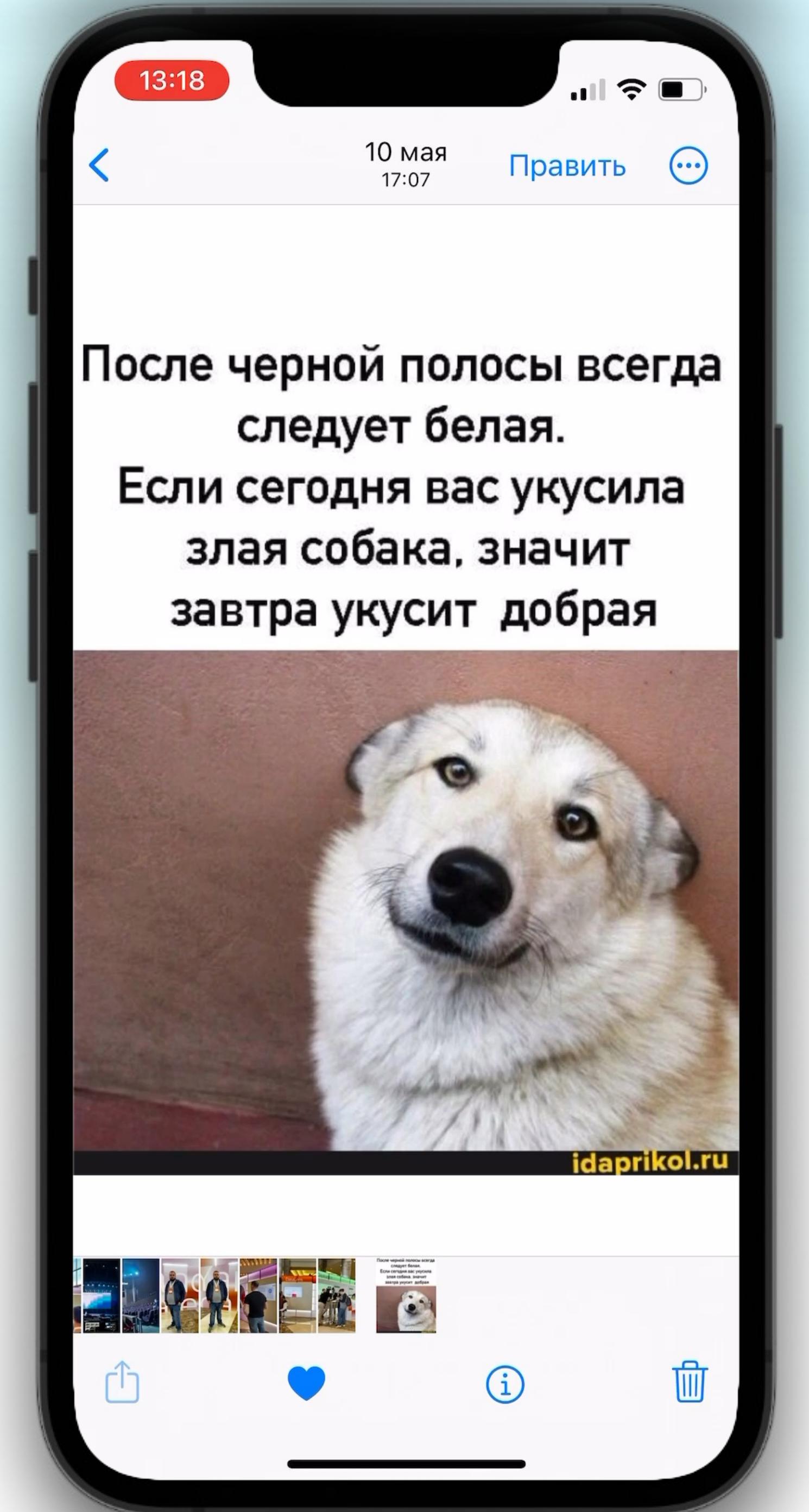
Documentation



To allow iOS to include conversations from your messaging app in the list of suggestions:

- Add a share extension to your app as described in [Understand Share Extensions](#).
- Declare support for the [`INSendMessageIntent`](#) intent type.
- Donate an [`INSendMessageIntent`](#) in your messaging app and its share extension.

Пара дней, и...



**Siri начинает работать на
личных телефонах**



Это доклад все еще про CarPlay

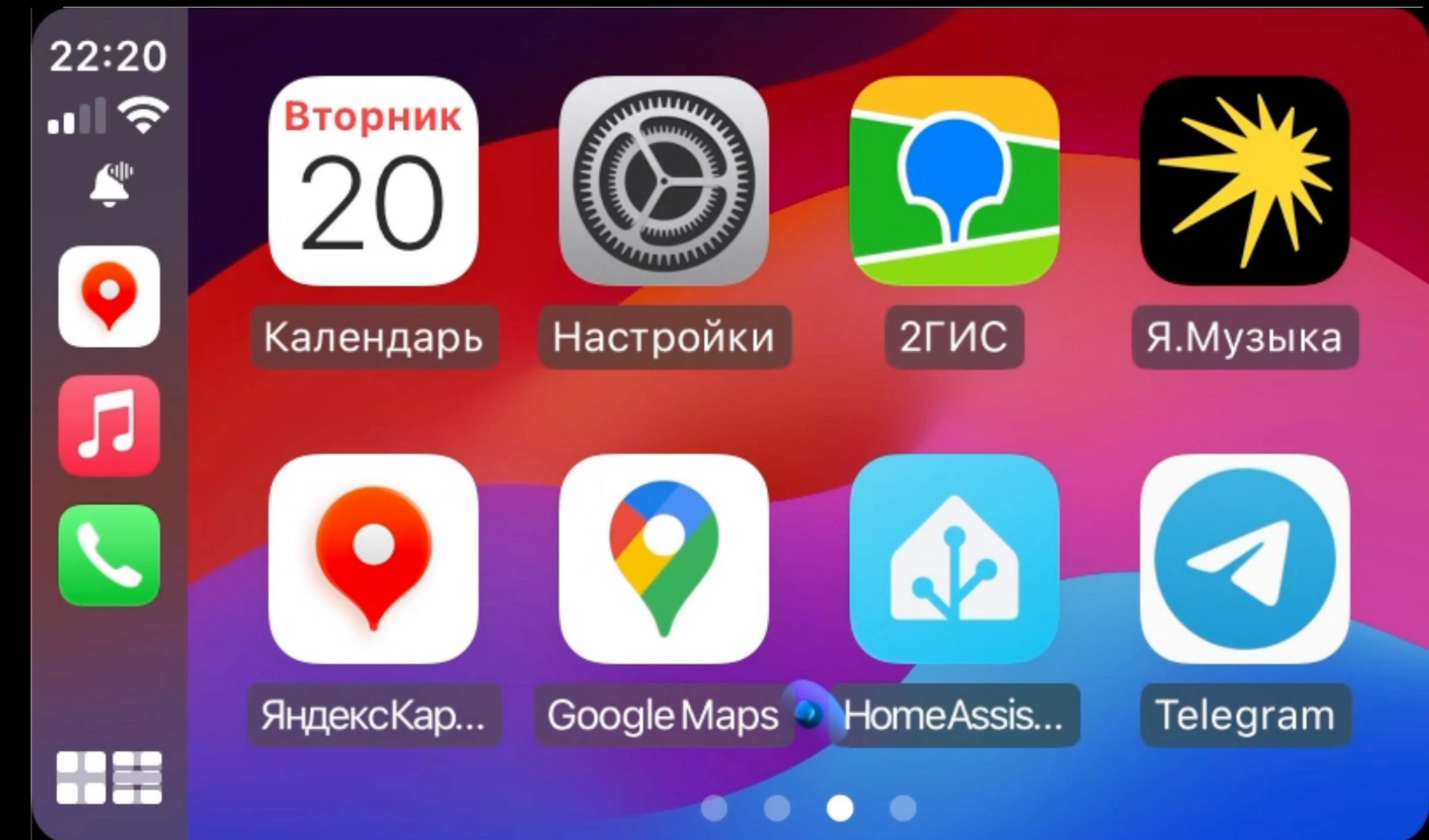
Как работает CarPlay?

Car

iPhone



Как работают другие мессенджеры?



BACK

CarPlay App Programming Guide

- Audio
- Communication (messaging and calling)
- Driving task
- EV charging
- Fueling
- Navigation (turn-by-turn directions)
- Parking
- Quick food ordering

- Audio
- Communication (messaging and calling)
- Driving task
- EV charging
- Fueling
- Navigation (turn-by-turn directions)
- Parking
- Quick food ordering

com.apple.developer.carplay-communication

Telegram
597a750

iPhone 15 Pro

Indexing | Processing files

+

Find > Text > Containing

Aa

CarPlay

In Project

21 results in 3 files

BUILD Telegram > Telegram

 official_carplay_fragment = ""

 <key>com.apple.developer.carplay-messaging</key>

 carplay_fragment = official_carplay_fragment if telegram_bundle_id in official_bundle_ids...

 carplay_fragment = official_carplay_fragment if telegram_bundle_id in official_bundle_ids...

 carplay_fragment,

ASTraitCollection Telegram > su...

 case UIUserInterfaceIdiomCarPlay:

 return @"CarPlay";

AppDelegate Telegram > submo...

 var authorizationOptions: UNA

433 <true/>

434 """

435 unrestricted_voip_fragment =
 official_unrestricted_voip_fragment **if**
 telegram_bundle_id **in** official_bundle_ids **else** ""

436

437 official_carplay_fragment = ""

438 <key>com.apple.developer.carplay-messaging</key>

439 <true/>

440 """

441 carplay_fragment = official_carplay_fragment **if**
 telegram_bundle_id **in** official_bundle_ids **else** ""

442

443 icloud_fragment = "" **if not** telegram_enable_icloud
 else ""

444 <key>com.apple.developer.icloud-services</key>

445 <array>

446 <string>CloudKit</string>

7 characters

Странно...

Communication apps can support the CarPlay framework in addition to SiriKit and CallKit. Be sure to include the correct entitlement(s) to match the frameworks and features you support.

If your app needs to work on iOS 13 and earlier, also include the

`com.apple.developer.carplay-messaging` and/or `com.apple.developer.carplay-calling` entitlements to match your app features. Apps that don't support the CarPlay framework will still work on later versions of iOS, but without a customized user interface.

All communication apps must support required SiriKit intents, and CallKit (for calling apps). For a list of required SiriKit intents for communication apps, see [CarPlay app guidelines](#).

Entitlement	Key
CarPlay Messaging App	<code>com.apple.developer.carplay-messaging</code>
Deprecated. App relies solely on SiriKit and supports SiriKit intents to send, request, and modify messages. May be combined with the CarPlay Communication App entitlement, and the optional CarPlay VoIP Calling App entitlement to support iOS 13 and earlier.	
CarPlay VoIP Calling App	<code>com.apple.developer.carplay-calling</code>
Deprecated. App relies solely on SiriKit and CallKit, and supports SiriKit intents for starting calls and requesting a list of calls. May be combined with the CarPlay Communication App entitlement, and the optional CarPlay Messaging App entitlement to support iOS 13 and earlier.	

**Как у других мы
сделать не можем...**

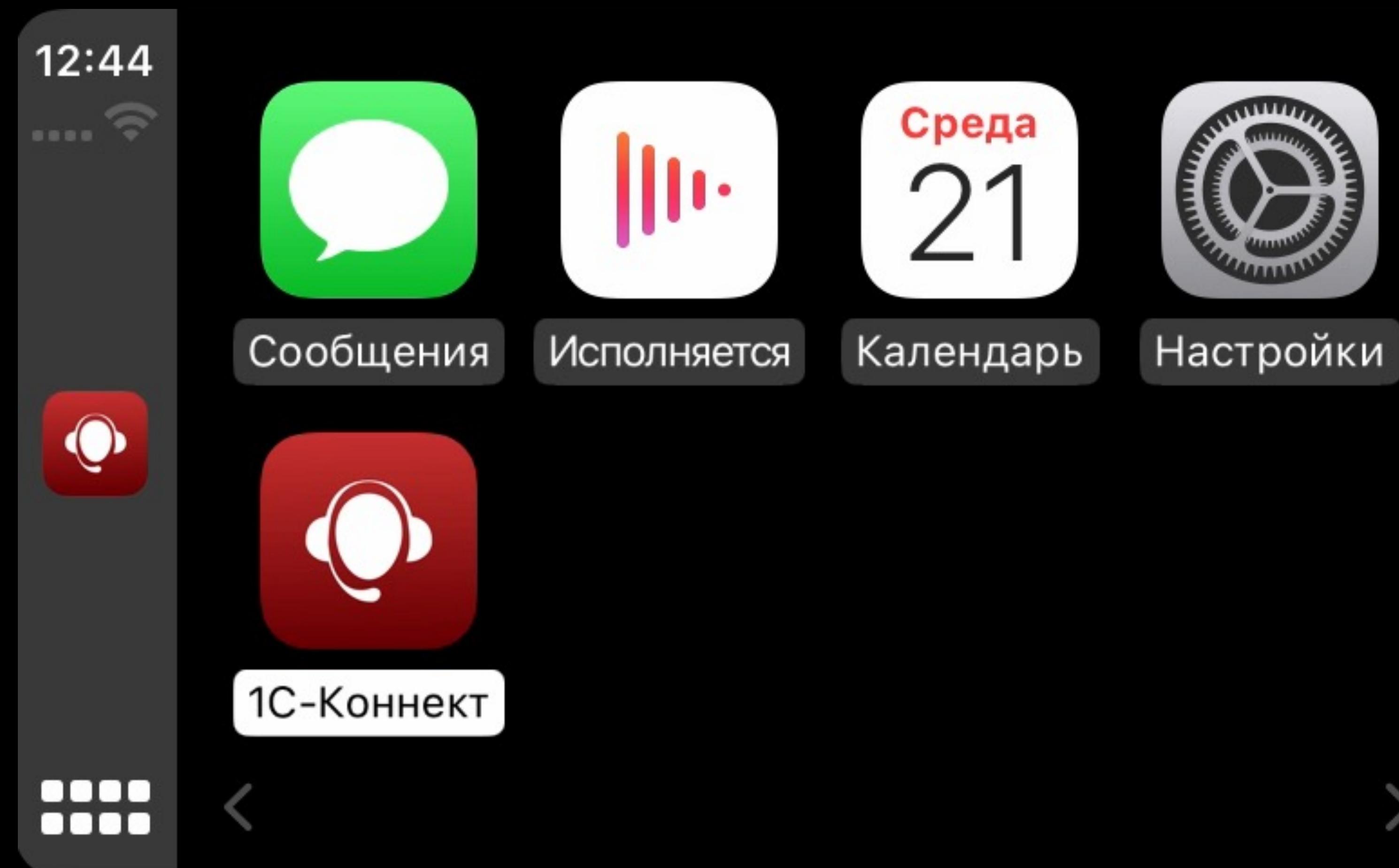
Ho...

Можем сделать лучше!

Driving task, EV
charging, fueling,
parking, and quick
food ordering

	Audio	Communication	Navigation	
Action Sheet	● *1	●	●	●
Alert	●	●	●	●
Grid	●	●	●	●
List	●	●	●	●
Tab bar	●	●	●	●
Information		●	●	●
Point of Interest				●
Now Playing	●	● *1		
Contact		●	●	
Map			●	
Search			●	
Voice control			●	

*1 New in iOS 17.



< BACK

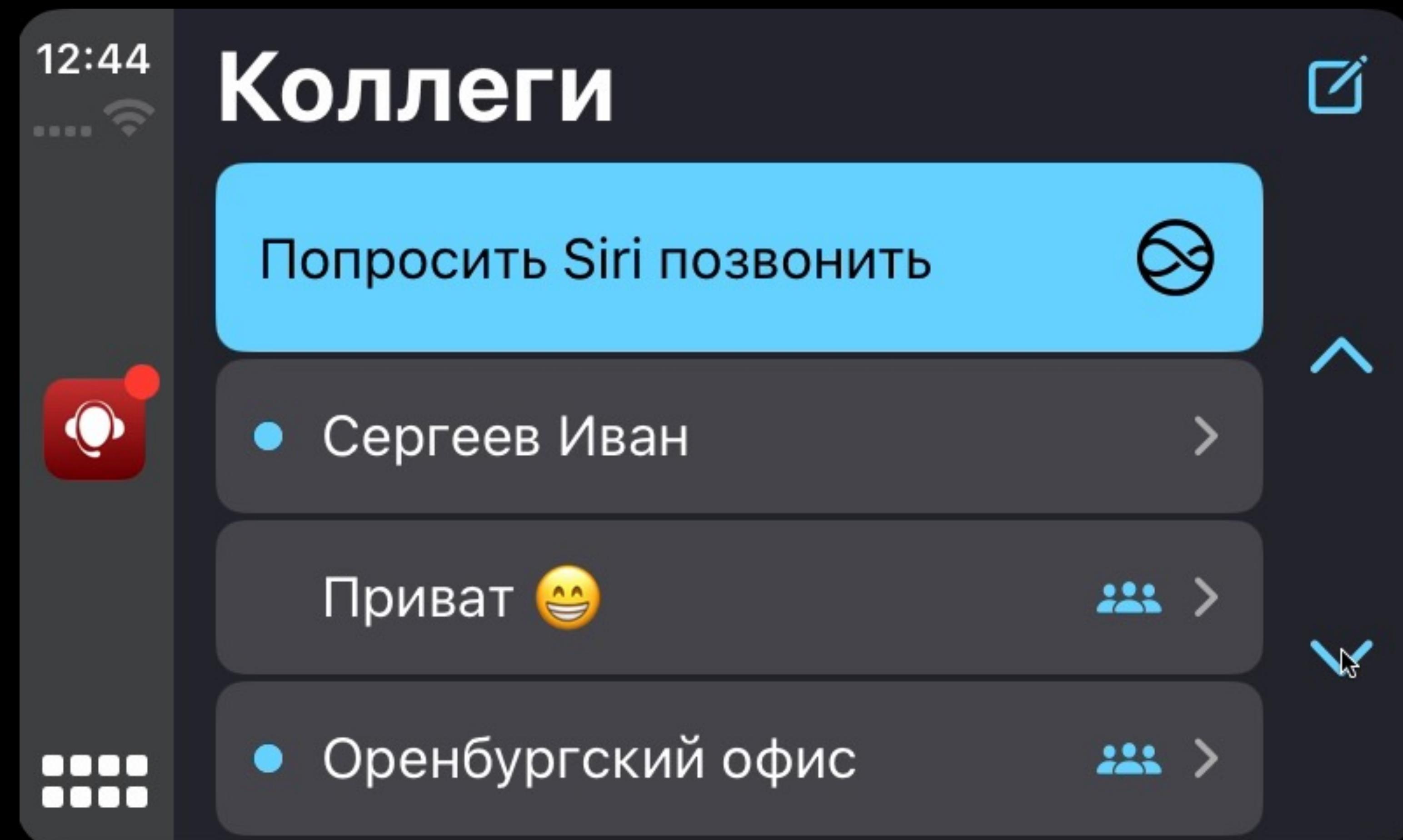
12:44



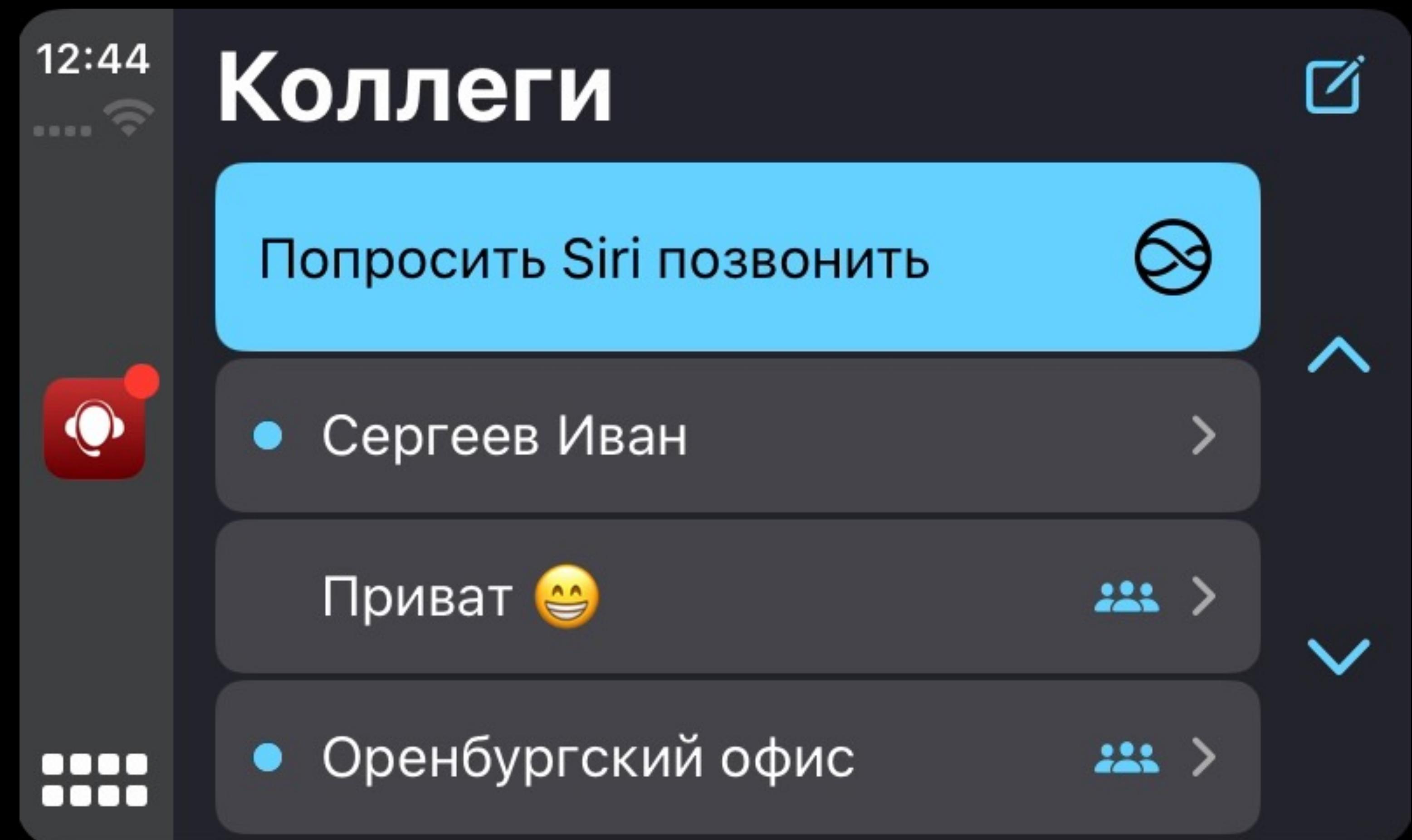
Остановите автомобиль
и войдите в свой аккаунт в 1С-
Коннект

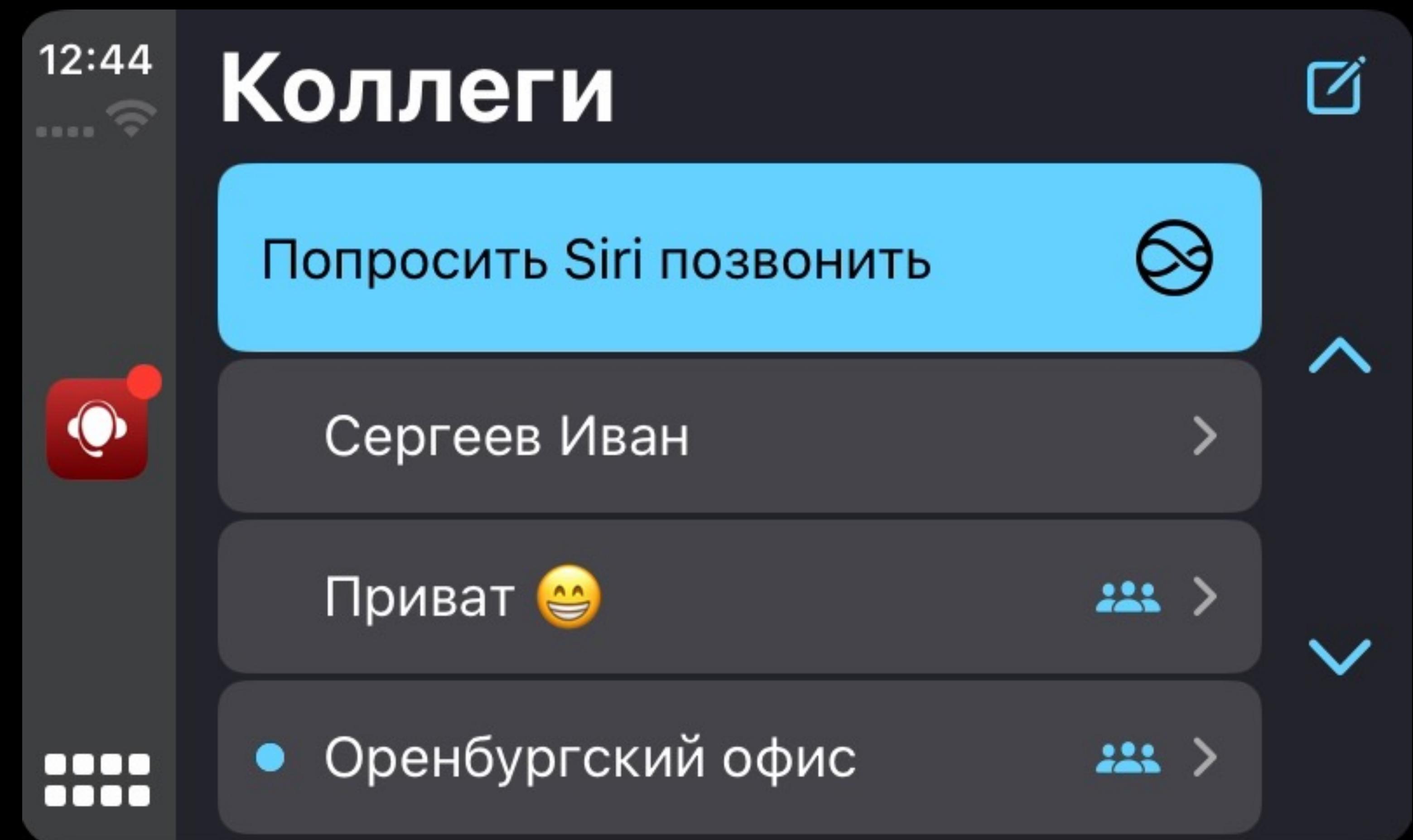


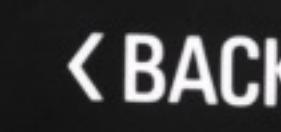
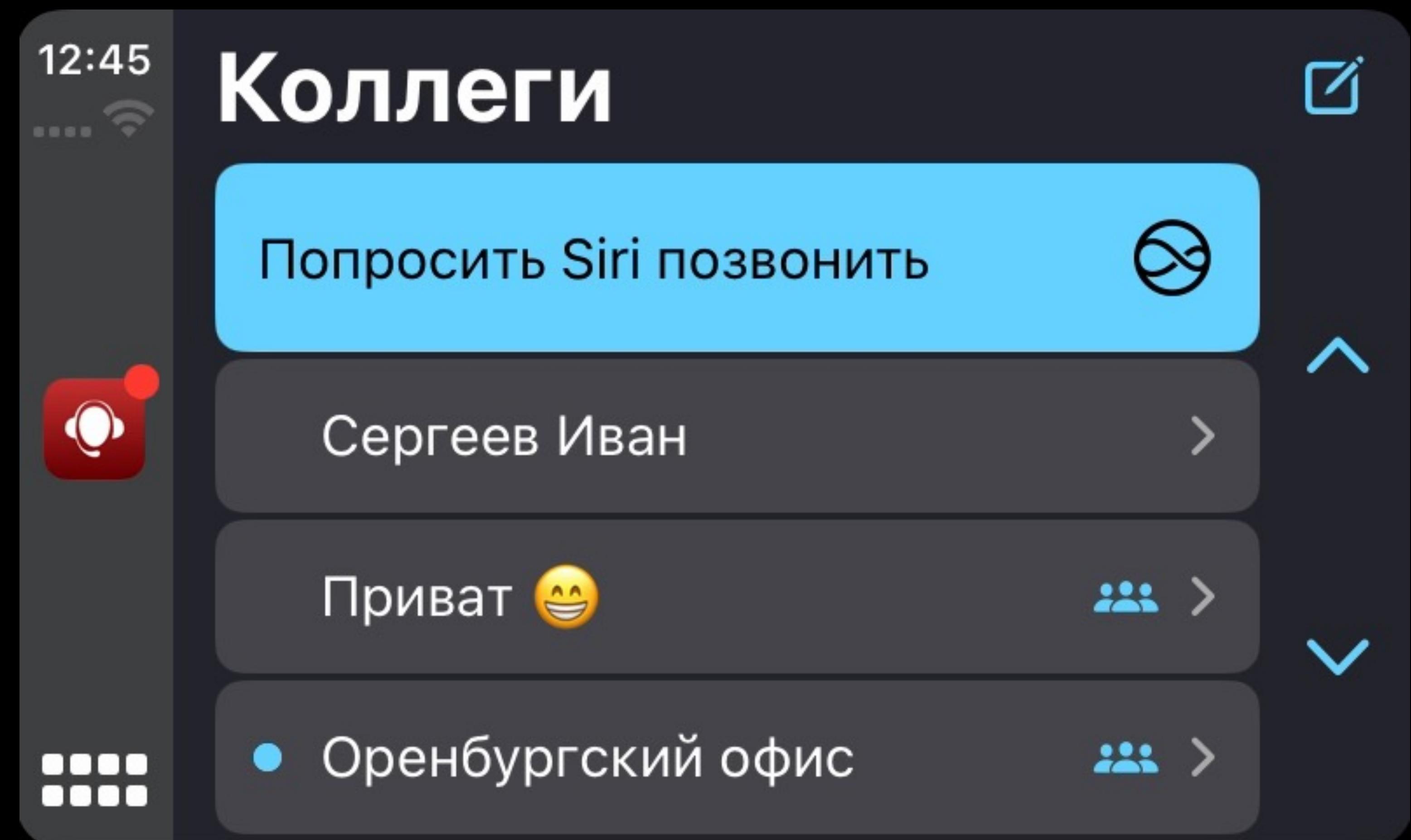
< BACK



< BACK







Как дебажить?





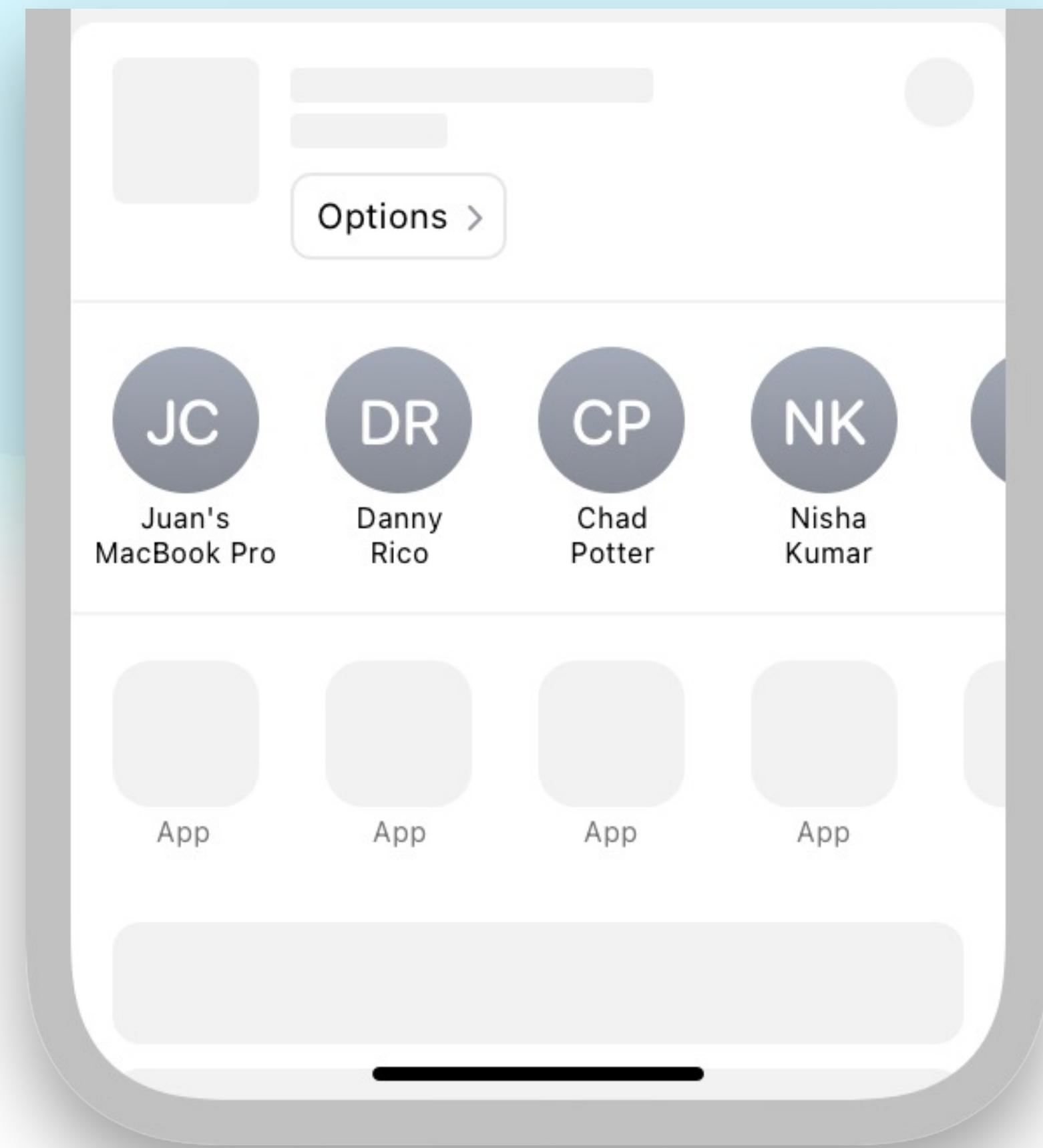
**Все круто!
CarPlay работает!**

Ведь работает?

Нет...

Почему?





Спасибо за внимание!

Вопросы?

Как мы (не)выпустили CarPlay



Сергей Долгих

iOS разработчик, 1С-Коннект



Материалы по докладу