

Яндекс Go

Яндекс **Go**

Яндекс GO готовит Backend Driven UI

Еркебулан Абилдин, iOS Разработчик

План

Познакомимся с Backend Driven UI

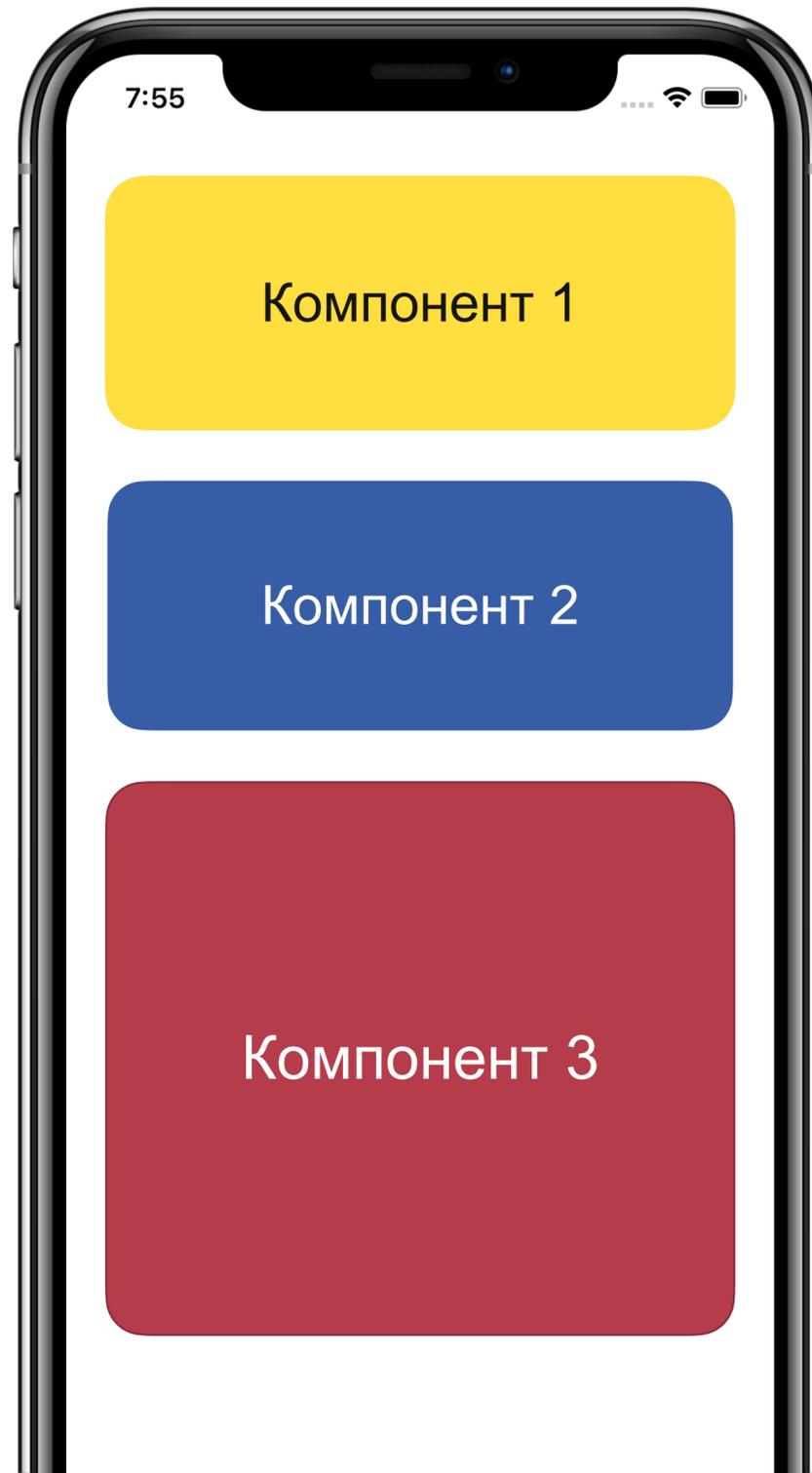
Разберем Backend Driven UI на примере шрифтов

Возможности Backend Driven UI

Обратная совместимость

Итоги

Backend Driven UI

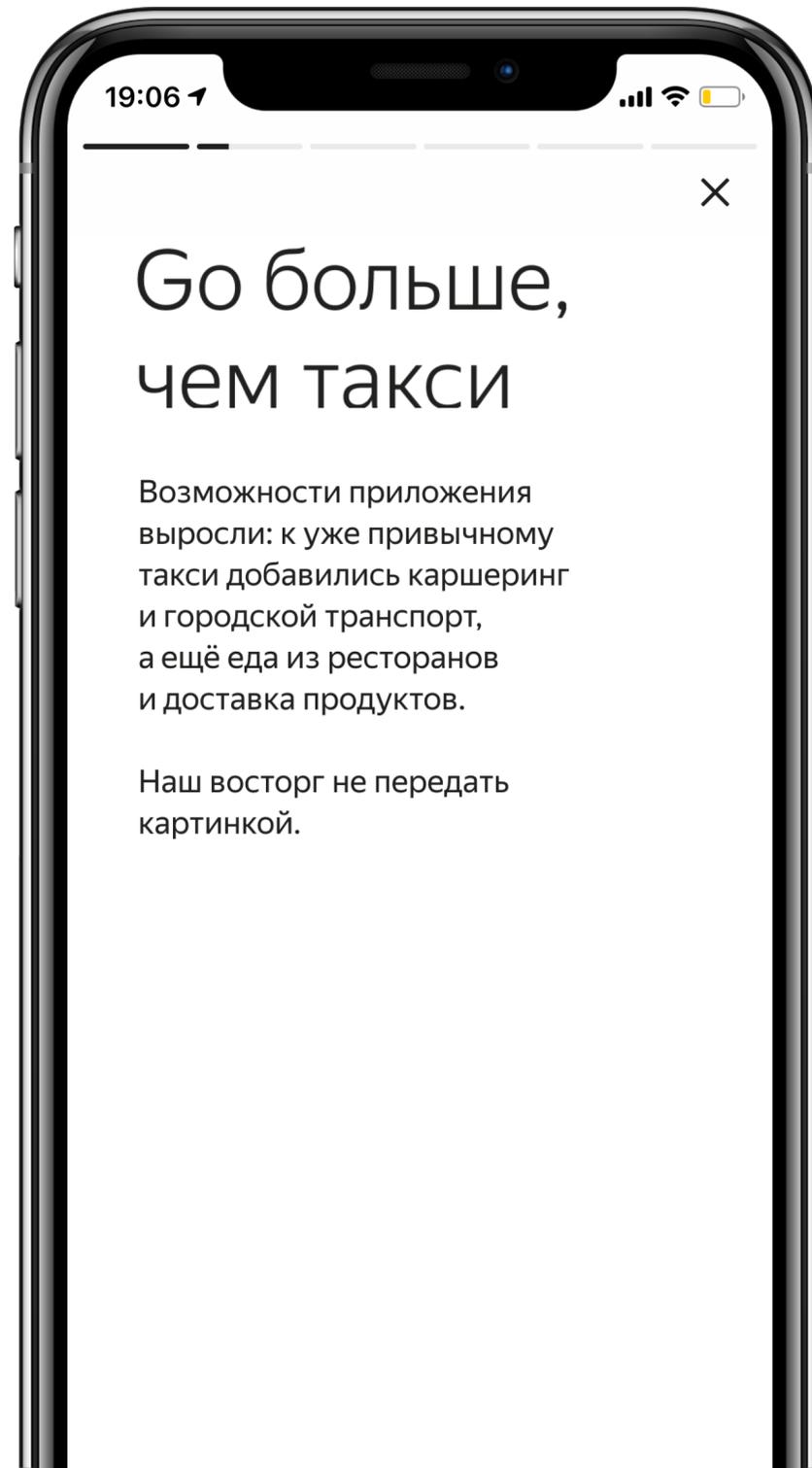


UI Построенный на основе ответа от backend

Список поддерживаемых компонентов заранее известен

Поддержка обратной совместимости

Продукт хочет изменить шрифт



Почему хотят это поменять?

- › Ребрендинг
- › Нужно поправить ошибку
- › Провести эксперимент
- › Нужно для промоушена
- › Просто хочется :)

Будет ли необходимость повторных изменений?

Разовое изменение:

- › Ребрендинг
- › Нужно поправить ошибку

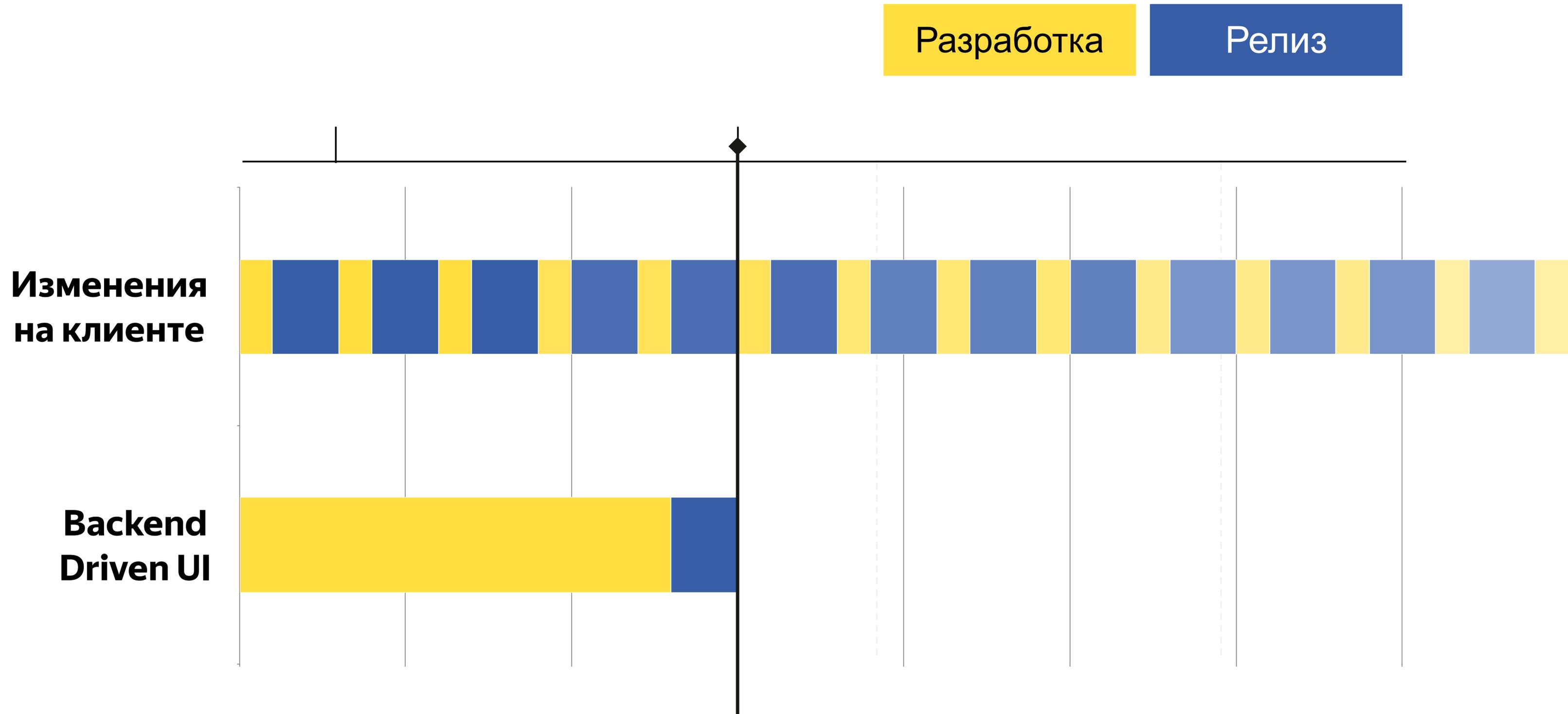
Есть вероятность дальнейших изменений:

- › Провести эксперимент
- › Нужно для маркетинга
- › Просто хочется :)

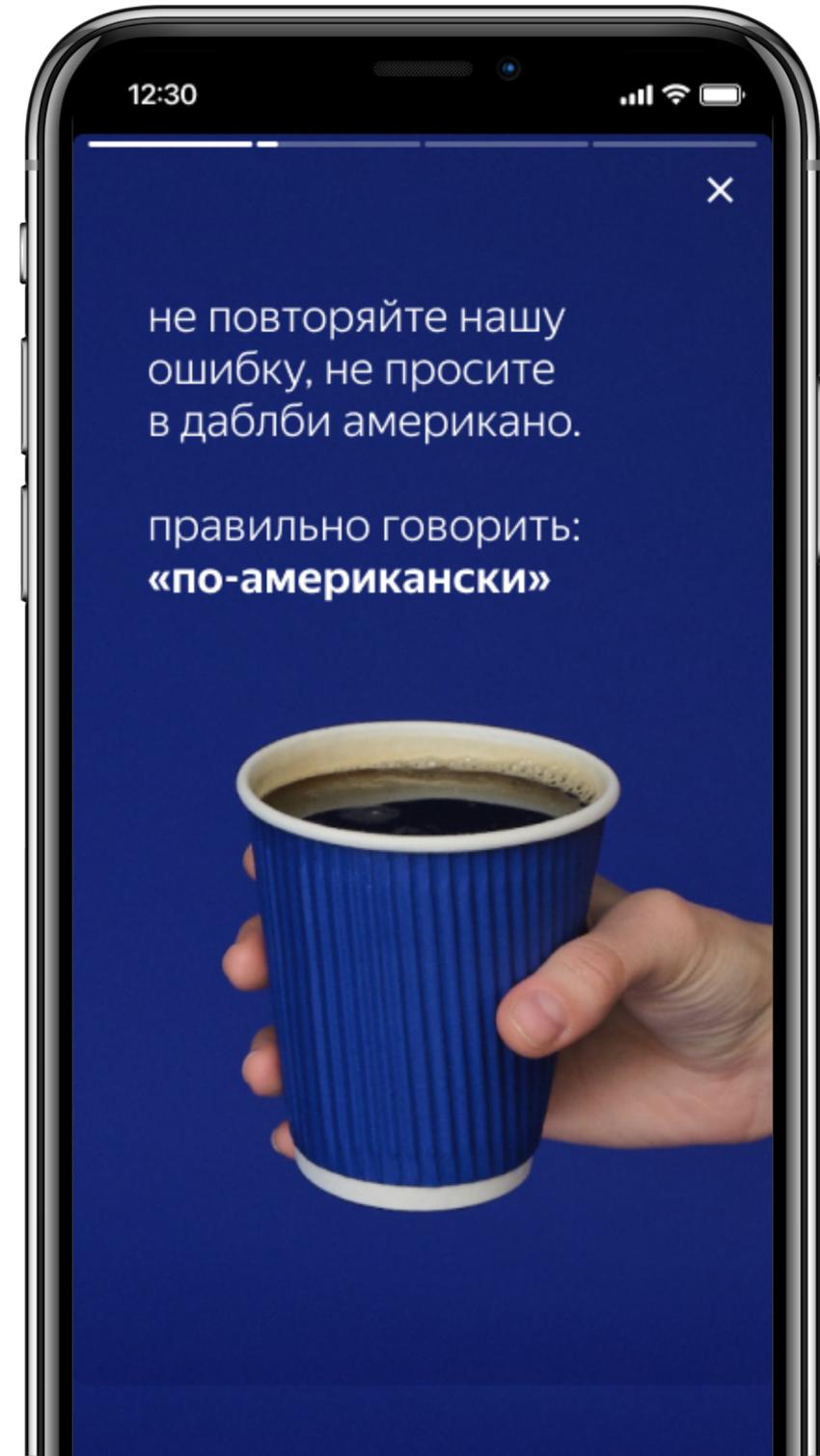
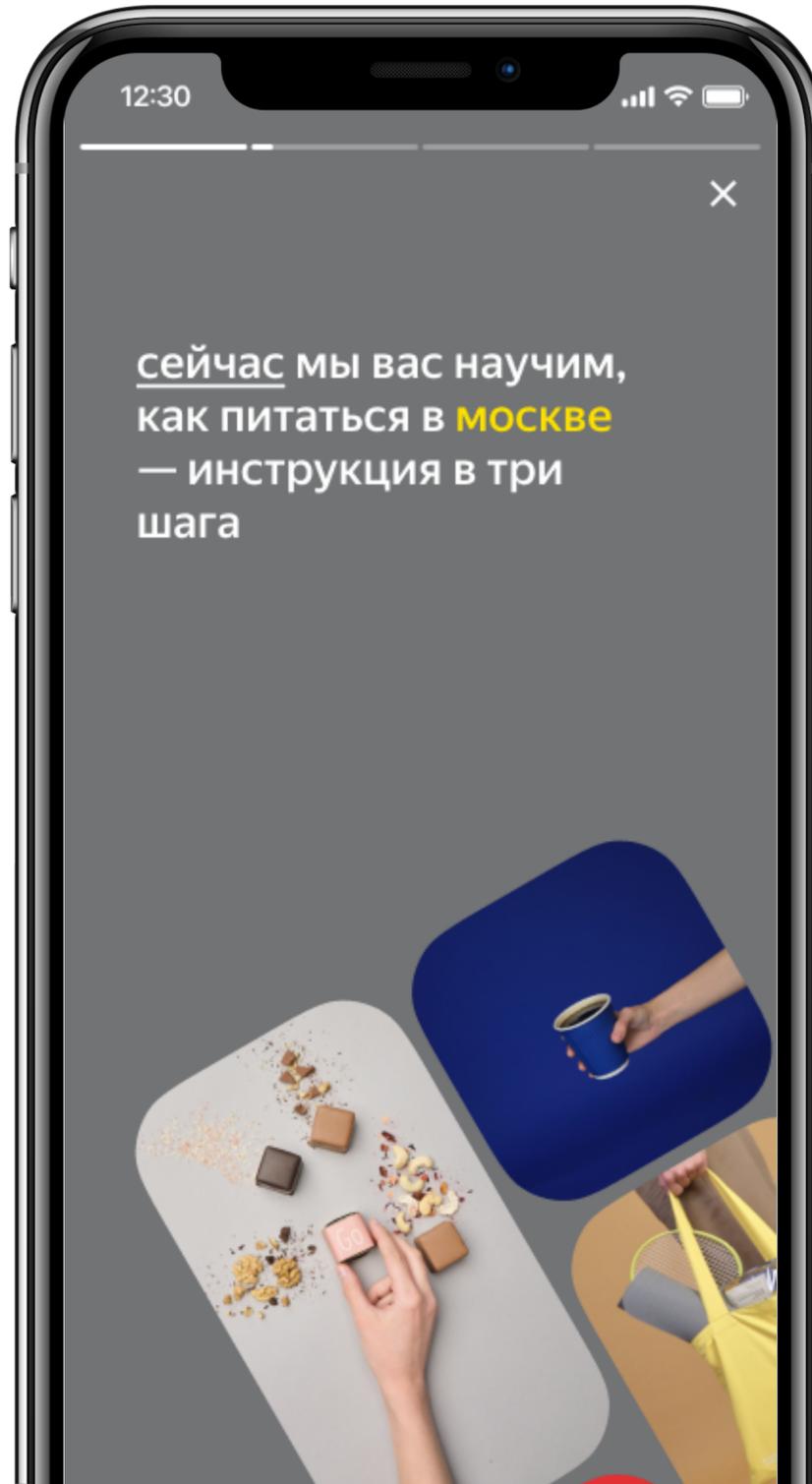
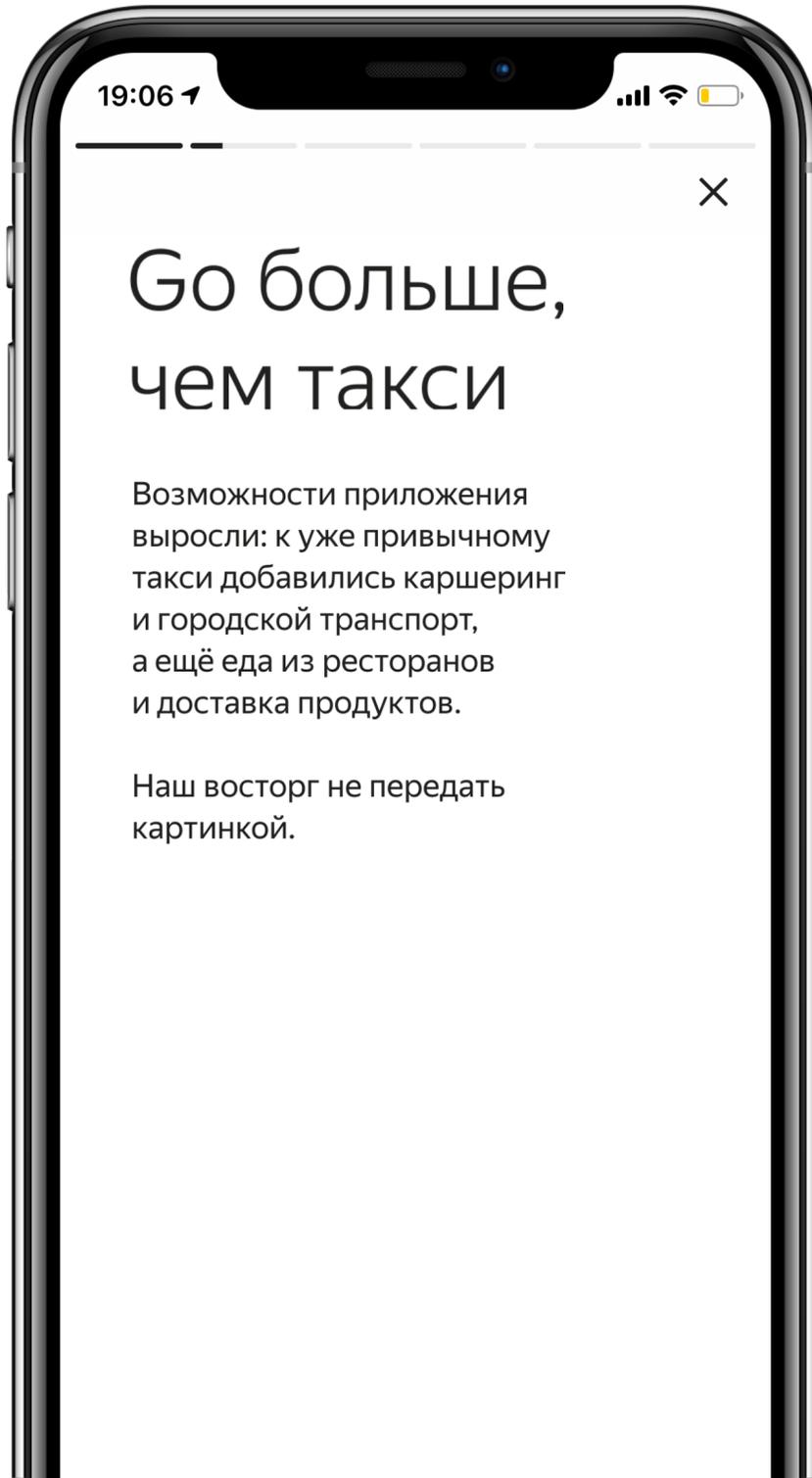
Что именно хотим контролировать с сервера?

- › Шрифт
- › Размер
- › Стиль
- › Цвет
- › Ссылки
- › Символы: Неразрывный пробел, перенос строки и тп

Цена реализации



Примеры использования



HTML разметка наше решение?

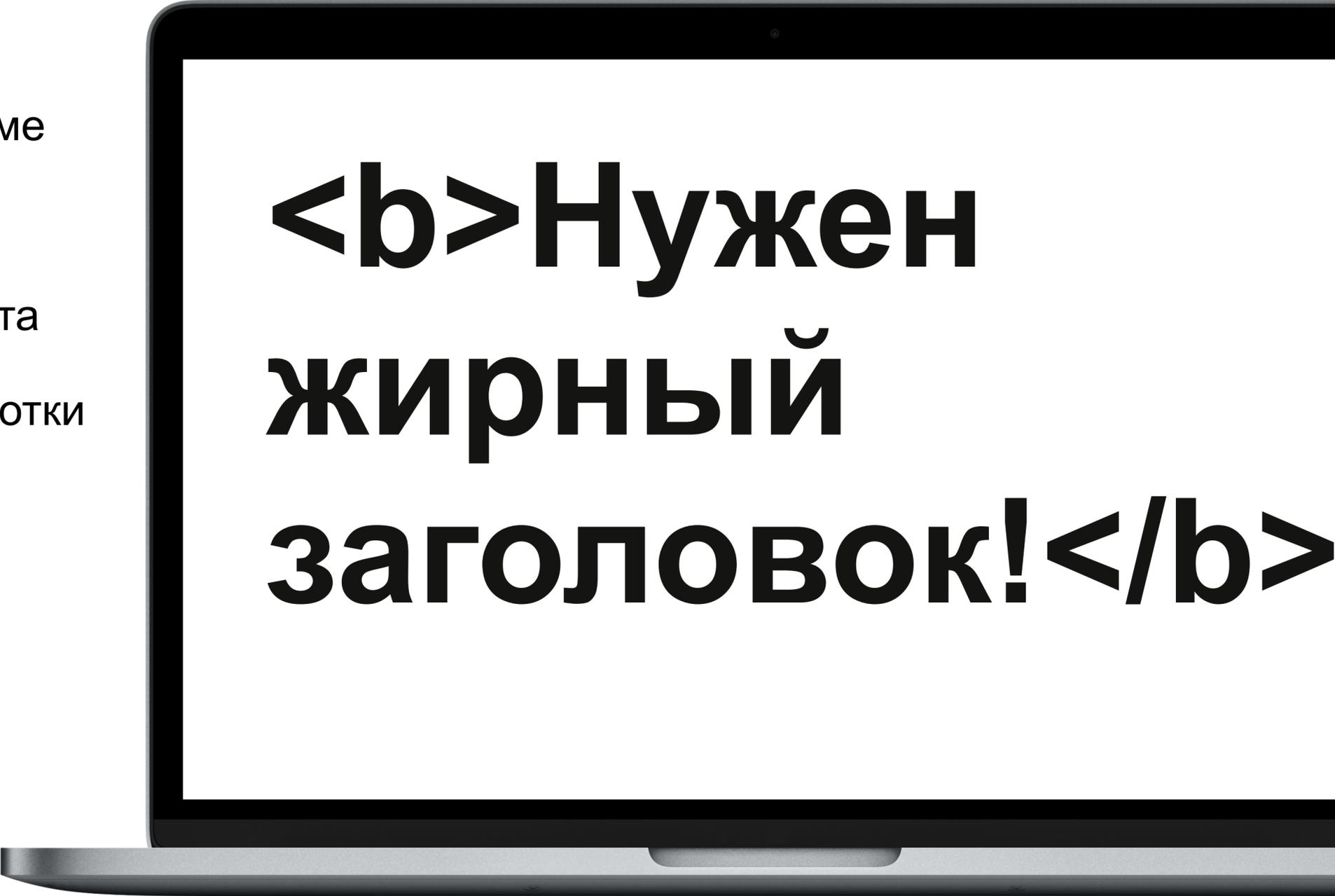
Текст станет очень гибким

Есть нативные конвертеры в платформе

С HTML знаком *почти* каждый

Логика тегов в тексте достаточно проста

Не требуется много времени на доработки



``Нужен
жирный
заголовок!``

Возможности HTML разметки и наши сильно отличались

- Не было понимания полного списка поддерживаемых возможностей

- То что поддерживалось на одной платформе, могло не поддерживаться на другой

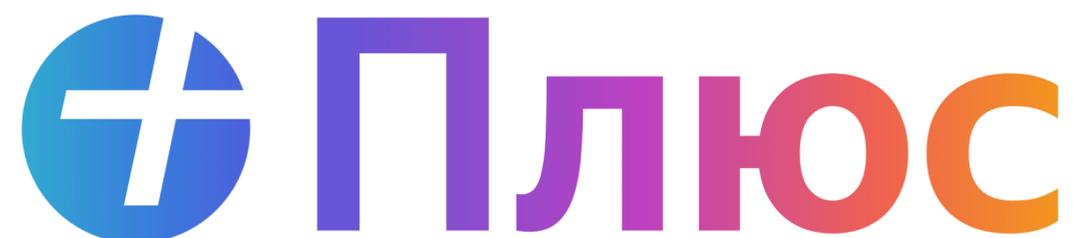
- Приходилось проверять контент на обеих платформах



Желание кастомизации сильно выросло

Появилась необходимость накладывать градиент вместо цвета

Добавлять кастомные картинки прямо в текст по типу эмодзи



NSAttributedString + HTML = Crash



I really need the info that such a report contains

Someone else sent me an Apple crash report and that let me refresh my understanding of this issue. Sadly, this is a known bug in iOS (r. 23592459) that can potentially affect anyone who constructs an



`NSAttributedString`

from HTML.

There isn't a good workaround for this other than to avoid this API altogether. My advice:

- If you're displaying large chunks of complex HTML, use a

`WKWebView`

- If this HTML is highly constrained — perhaps you're just using HTML as an easy way to transfer a constrained set of attributes, like bold and italics — create your own markup system that doesn't rely on HTML. Or parse the HTML for just these attributes and use the result to create your attributed string.

I'm sorry I don't have better news here.

Share and Enjoy — Quinn "The Eskimo!" Apple Developer Relations, Developer Technical Support, Core OS/Hardware

```
let myEmail = "eskimo" + "1" + "@apple.com"
```

Posted 1 year ago by [eskimo](#)  



HTML конвертируется не за 1 итерацию RunLoop

```
func convertHTML() {
    do {
        let html = "Text <br> <b>Bold Text</b>"
        let htmlData: Data! = html.data(using: .utf8)
        let result = try NSAttributedString(
            data: htmlData,
            options: [.documentType: NSAttributedString.DocumentType.html],
            documentAttributes: nil
        )
        print("html length: \(html.count). converted result length: \(result.length)")
    } catch {
        print("error: \(error.localizedDescription)")
    }
}
```

```
func runGCDTest() {  
    print("1")  
    DispatchQueue.main.async {  
        print("4")  
    }  
    print("2")  
    convertHTML()  
    print("3")  
    DispatchQueue.main.async {  
        print("5")  
    }  
}
```

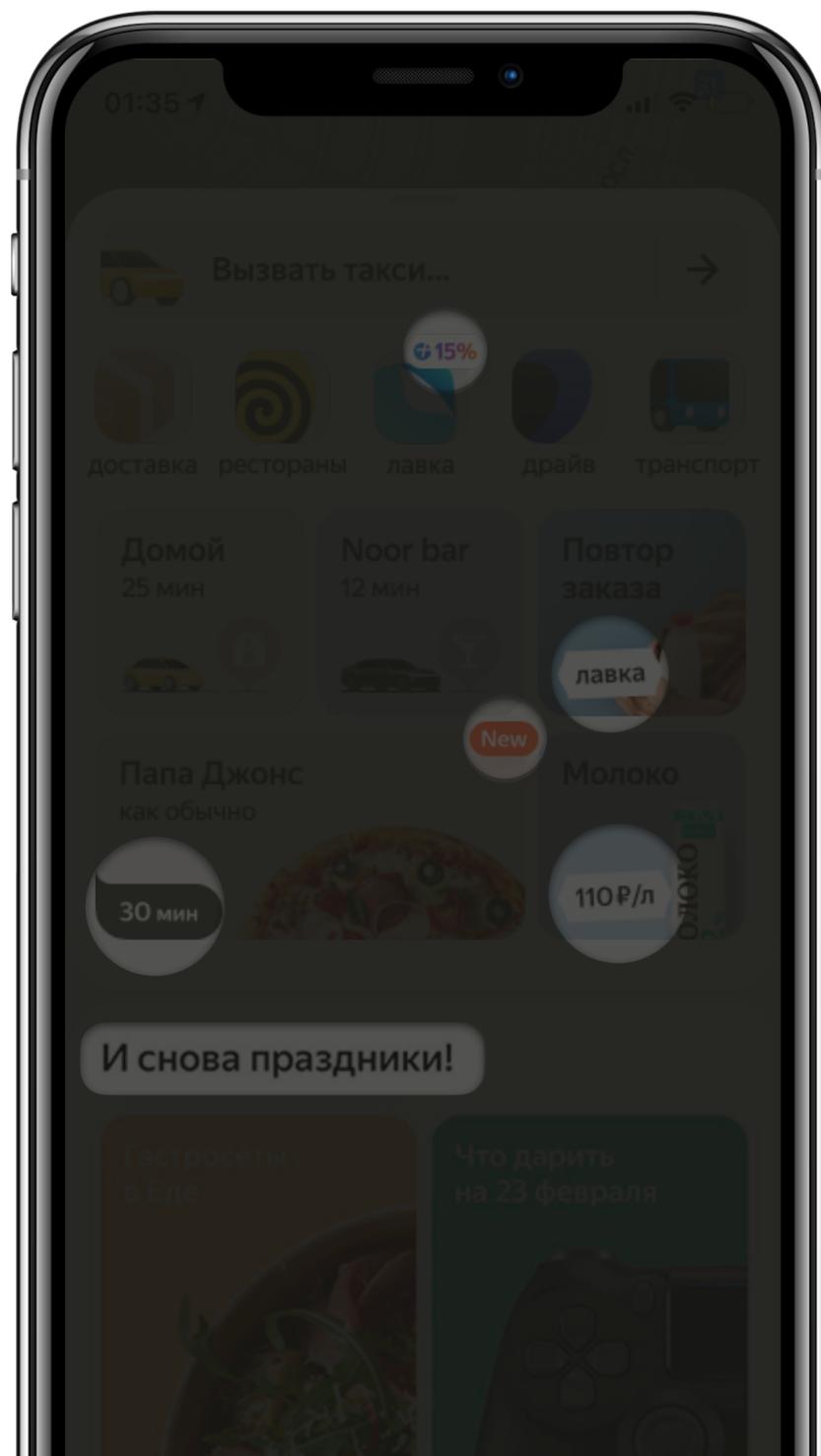
Вызываем с **main** по нажатию кнопки

Ожидаемый результат: **1, 2, (html output), 3, 4, 5**

Полученный результат: **1, 2, 4, (html output), 3, 5**



Новый сценарий использования



Отказываемся от HTML

Нужен новый подход, покрывающий все необходимые кейсы

Смотрим на текст с точки зрения Backend Driven UI

[Картинка]

[Текст]

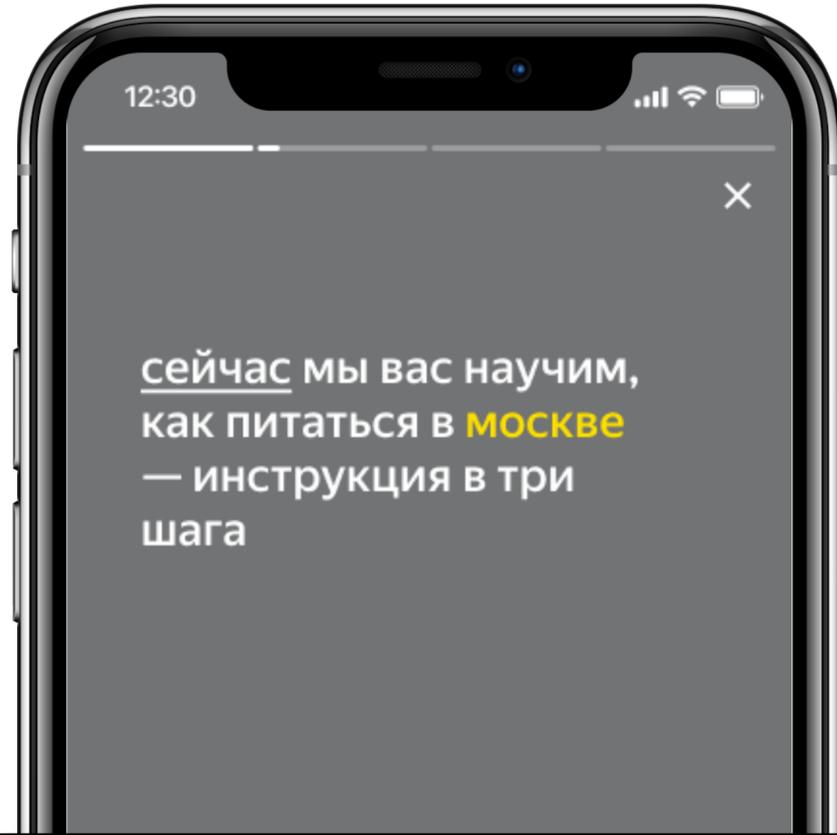
[Ссылка]

- › url: ya.ru/img/123
- › Высота: 110
- › Цвет (optional)

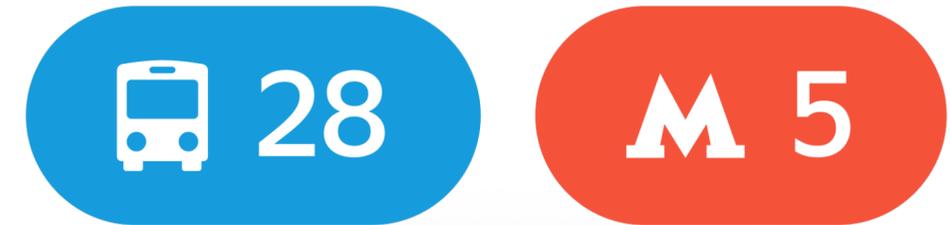
- › Цвет: Градиент
- › Стиль: Жирный
- › Размер: 110
- › Шрифт: Arial

- › Цвет: Синий
- › Стиль: Курсив
- › Размер: 110
- › Шрифт: Arial
- › Action.url: ya.ru

Примеры использования



[текст][текст]
[текст][текст]



[картинка][текст]

План

Познакомились с Backend Driven UI

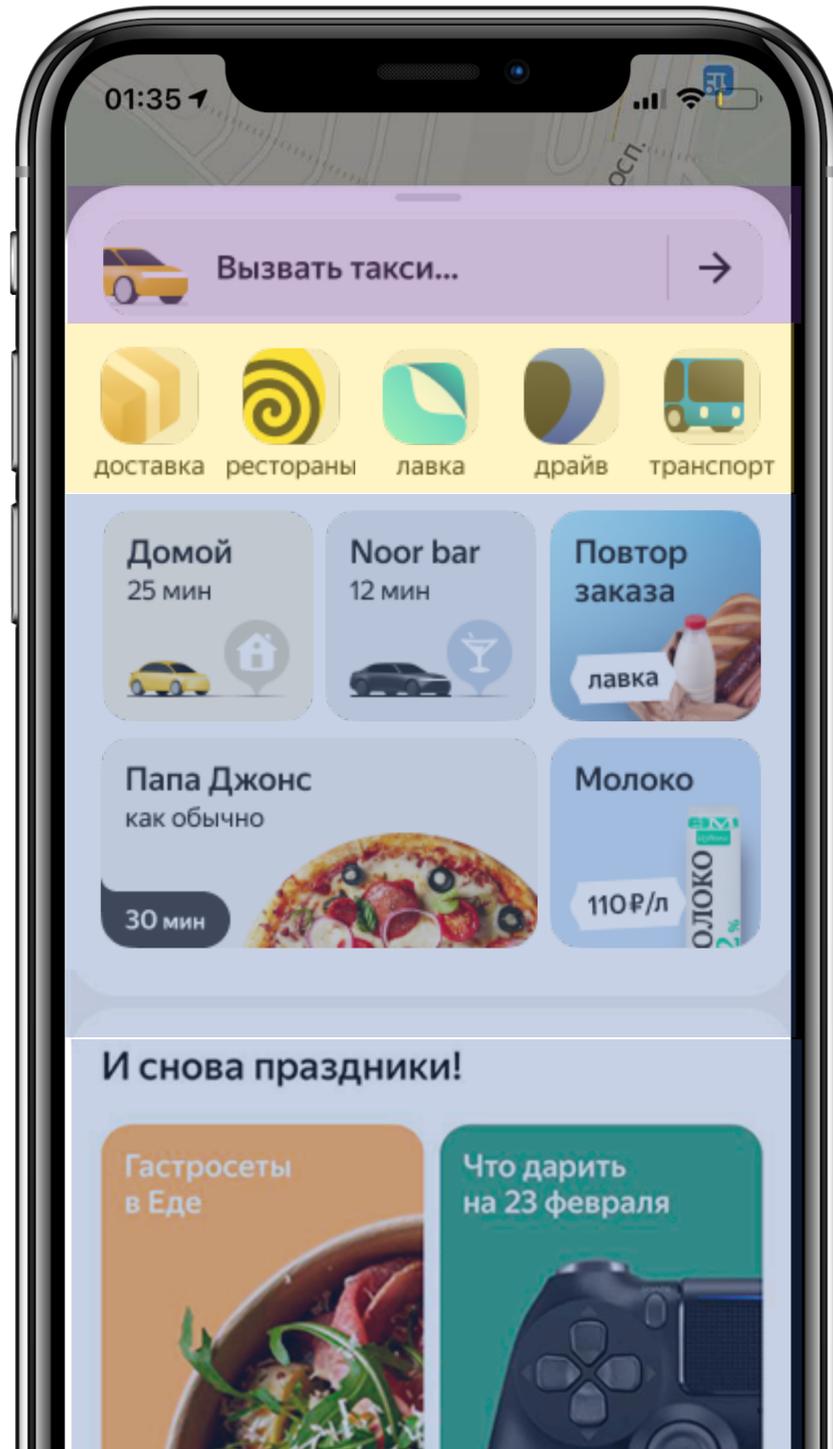
Разобрали Backend Driven UI на примере шрифтов

Возможности Backend Driven UI

Обратная совместимость

Итоги

Штора Яндекс Go

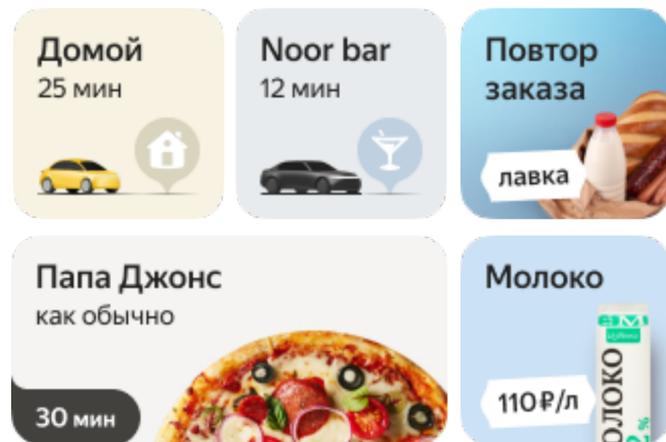
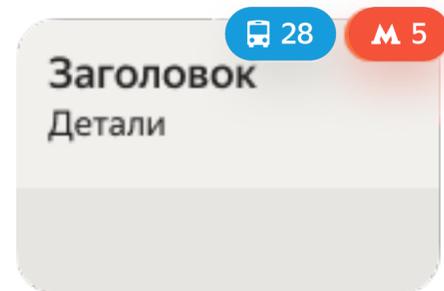


Штора состоит из больших компонентов

Компоненты содержат другие компоненты внутри себя

Используются «Компонентные текста»

Описание шортката



Размер и расположение на сетке

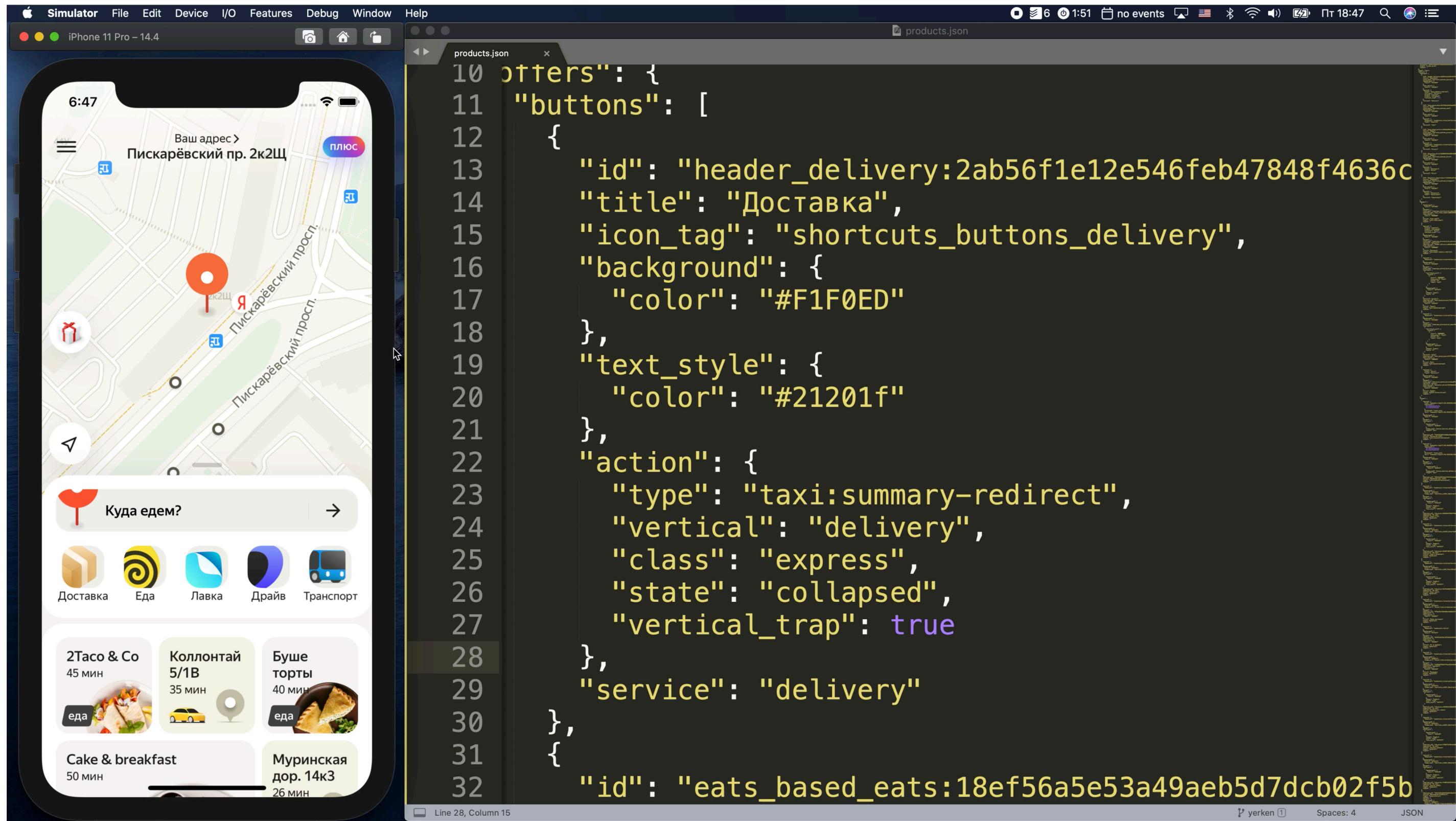
Визуальное описание

- › Компонентные текста
- › *Сверху крепятся бейджи, органичные лишь длиной компонента*
- › *Footer содержит картинку тарифа, текст или плашку*
- › *Background: картинка или цвет*

Функциональное описание

- › *Действие по нажатию*
- › *Отображение информации сервиса: ETA, Статус заказа и т.д.*

Изменение параметров Action

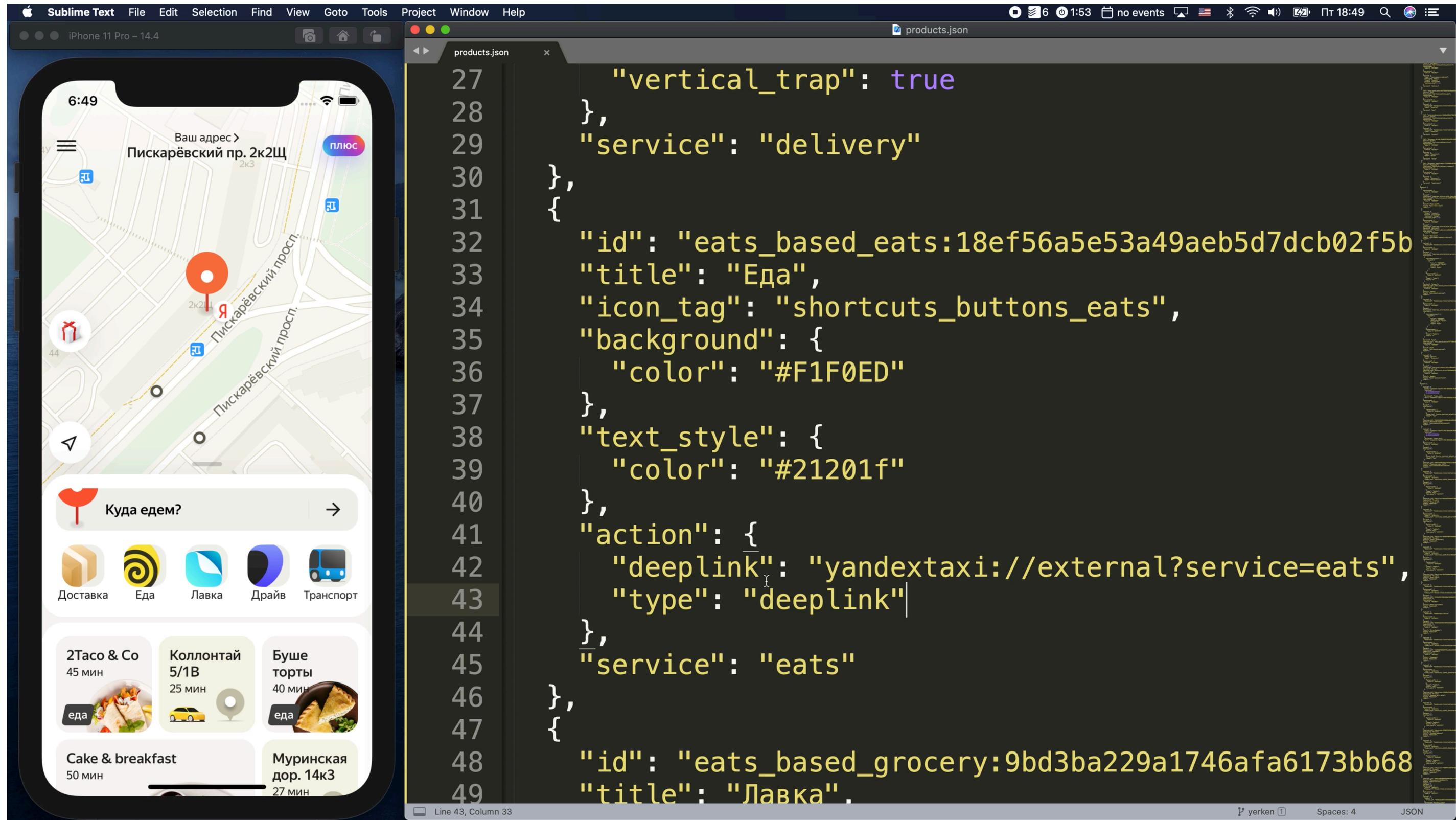


The image shows a side-by-side comparison of a mobile application interface and its underlying JSON configuration. On the left, an iPhone 11 Pro simulator displays a map with a red location pin and a bottom navigation bar. The bottom bar includes a search field "Куда едем?" and several service icons: "Доставка" (Delivery), "Еда" (Food), "Лавка" (Store), "Драйв" (Drive), and "Транспорт" (Transport). Below the icons are several food delivery cards, such as "2Тасо & Со" (45 min), "Коллонтай" (35 min), "Буше торты" (40 min), "Cake & breakfast" (50 min), and "Муринская дор." (26 min).

On the right, a code editor displays the JSON configuration for the "Доставка" button. The configuration is as follows:

```
10 "offers": {
11   "buttons": [
12     {
13       "id": "header_delivery:2ab56f1e12e546feb47848f4636c",
14       "title": "Доставка",
15       "icon_tag": "shortcuts_buttons_delivery",
16       "background": {
17         "color": "#F1F0ED"
18       },
19       "text_style": {
20         "color": "#21201f"
21       },
22       "action": {
23         "type": "taxi:summary-redirect",
24         "vertical": "delivery",
25         "class": "express",
26         "state": "collapsed",
27         "vertical_trap": true
28       },
29       "service": "delivery"
30     },
31   ],
32   "id": "eats_based_eats:18ef56a5e53a49aeb5d7dcb02f5b"
```

Замена Action



The image shows a side-by-side comparison between a mobile application interface and its underlying JSON data. On the left, an iPhone 11 Pro simulator displays a map with a red location pin and a bottom panel with service cards for 'Доставка', 'Еда', 'Лавка', 'Драйв', and 'Транспорт'. On the right, a code editor shows the JSON structure for these services, with line 43 highlighted.

```
27     "vertical_trap": true
28   },
29   "service": "delivery"
30 },
31 {
32   "id": "eats_based_eats:18ef56a5e53a49aeb5d7dcb02f5b",
33   "title": "Еда",
34   "icon_tag": "shortcuts_buttons_eats",
35   "background": {
36     "color": "#F1F0ED"
37   },
38   "text_style": {
39     "color": "#21201f"
40   },
41   "action": {
42     "deeplink": "yandex:taxi://external?service=eats",
43     "type": "deeplink"
44   },
45   "service": "eats"
46 },
47 {
48   "id": "eats_based_grocery:9bd3ba229a1746afa6173bb68",
49   "title": "Лавка",
```


Изменение порядка

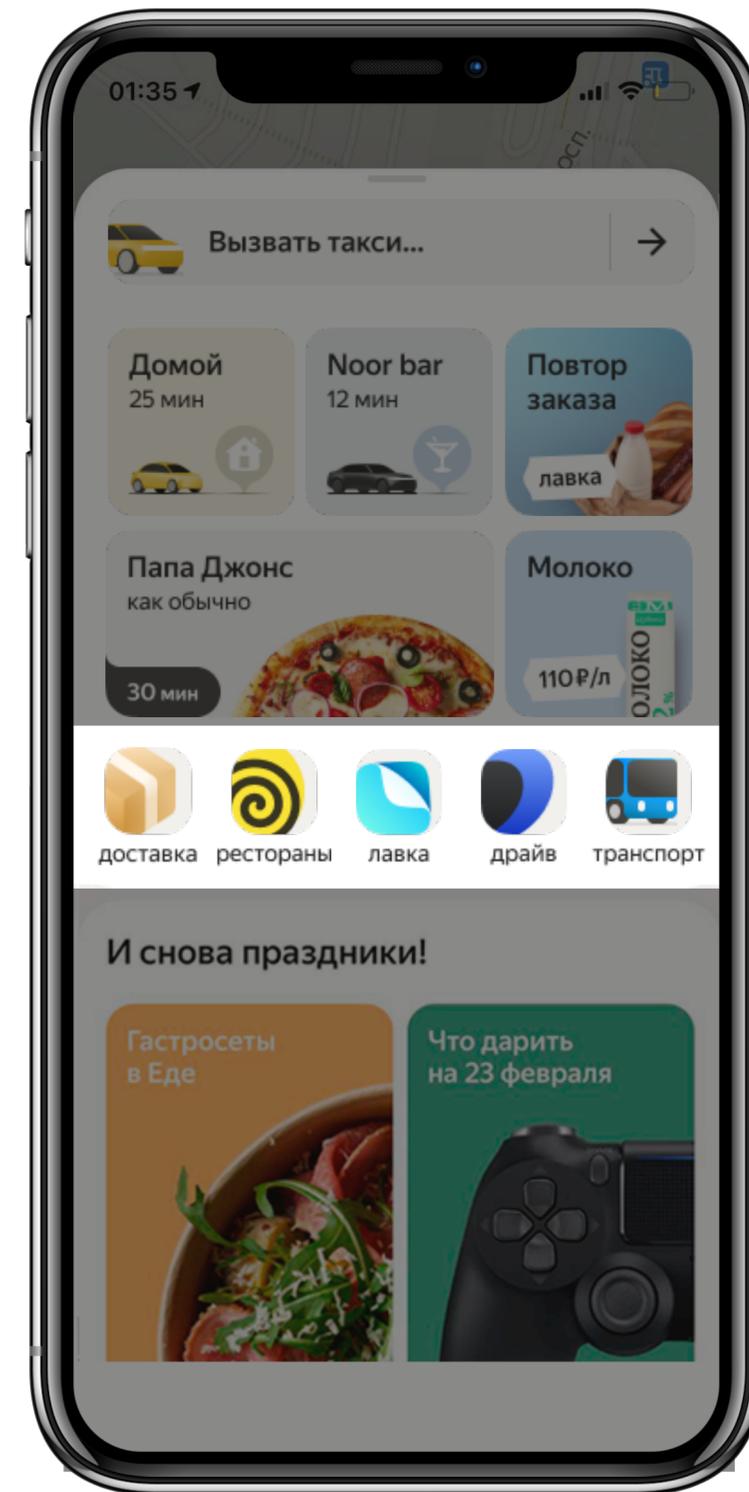
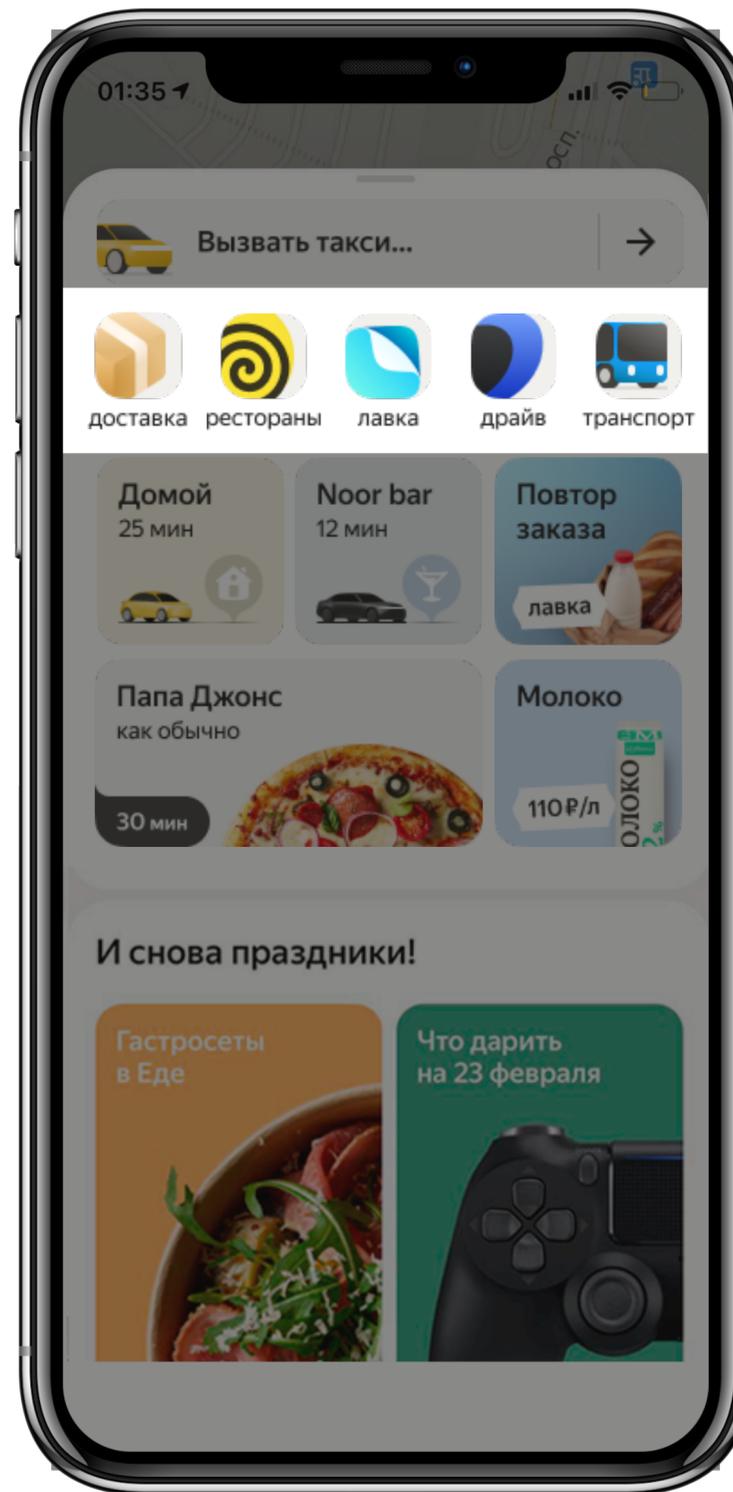
The image shows a side-by-side comparison of a mobile application's UI and its underlying data structure. On the left, an iPhone 11 Pro simulator displays a map application. The current location is 'Пискаревский пр. 2к2Щ'. Below the map is a search bar 'Куда едем?' and a row of service icons: 'Доставка', 'Еда', 'Лавка', 'Драйв', and 'Транспорт'. At the bottom, there are recommendations for food and services like '2Тасо & Со', 'Коллонтай', 'Буше торты', 'Cake & breakfast', and 'Муринская дор. 14к3'.

On the right, a code editor window titled 'products.json' shows a JSON array of objects. A search bar at the bottom of the editor contains the text 'button', and the results show 11 matches. The visible JSON code includes:

```
704     "eats_based_eats": "18e156a5e53a49aeb5d7dcb02150b0e1",
705     "eats_based_grocery": "9bd3ba229a1746afa6173bb6872a25",
706     "discovery_drive": "39a0b687b65a498fa2d38989093410c6",
707     "discovery_masstransit": "77c246d41e074a2d9ec7ec3df4",
708   ],
709   "tags": [
710     "gray_separator"
711   ]
712 },
713 },
714   "shortcut_ids": [
715     "eda-place-669186347bb64f70a1119b95f48d9064",
716     "7bbf5427d6f14396b1a04a5651687fe1",
717     "eda-place-d3dd9f4087434d63bcb05a423e23518b",
718     "eda-place-62e72ad6bbf04b389e750a6669211807",
719     "d0318ab607c84c17b7b1f113be20c152"
720   ],
721   "tags": [
722     "gray_separator"
723   ],
724   "type": "items_linear_grid"
725 }
```

А/Б Тестирование без изменений на Клиенте

Backend Driven UI дает возможность для проведения огромного количества А/Б тестов, не дожидаясь релизных циклов и раскатки на пользователей



A/B Тестирование на клиенте



Необходимо минимум 2 релиза

Часто код остается и после эксперимента

Эксперимент могут оставить в качестве конфига

Усложнение кода

Эксперименты + Backend Driven UI

Код клиента не меняется для эксперимента

Не нужно выпускать обновление

Код становится чище: без лишних ветвлений логики

Не будет устаревших флажков и экспериментов, знания о которых утеряны

После окончания эксперимента

Не нужно лезть в код для выпиливания эксперимента

Есть возможность подстраиваться под разные выборки пользователей

Можно улучшать с помощью ML

Эксперименты проводившиеся на Шорткатах

Зеркалили штору лево/право и вверх/вниз

Меняли местами Еду / Лавку / Доставку

Меняли рестораны предлагаемые ML на рандомные

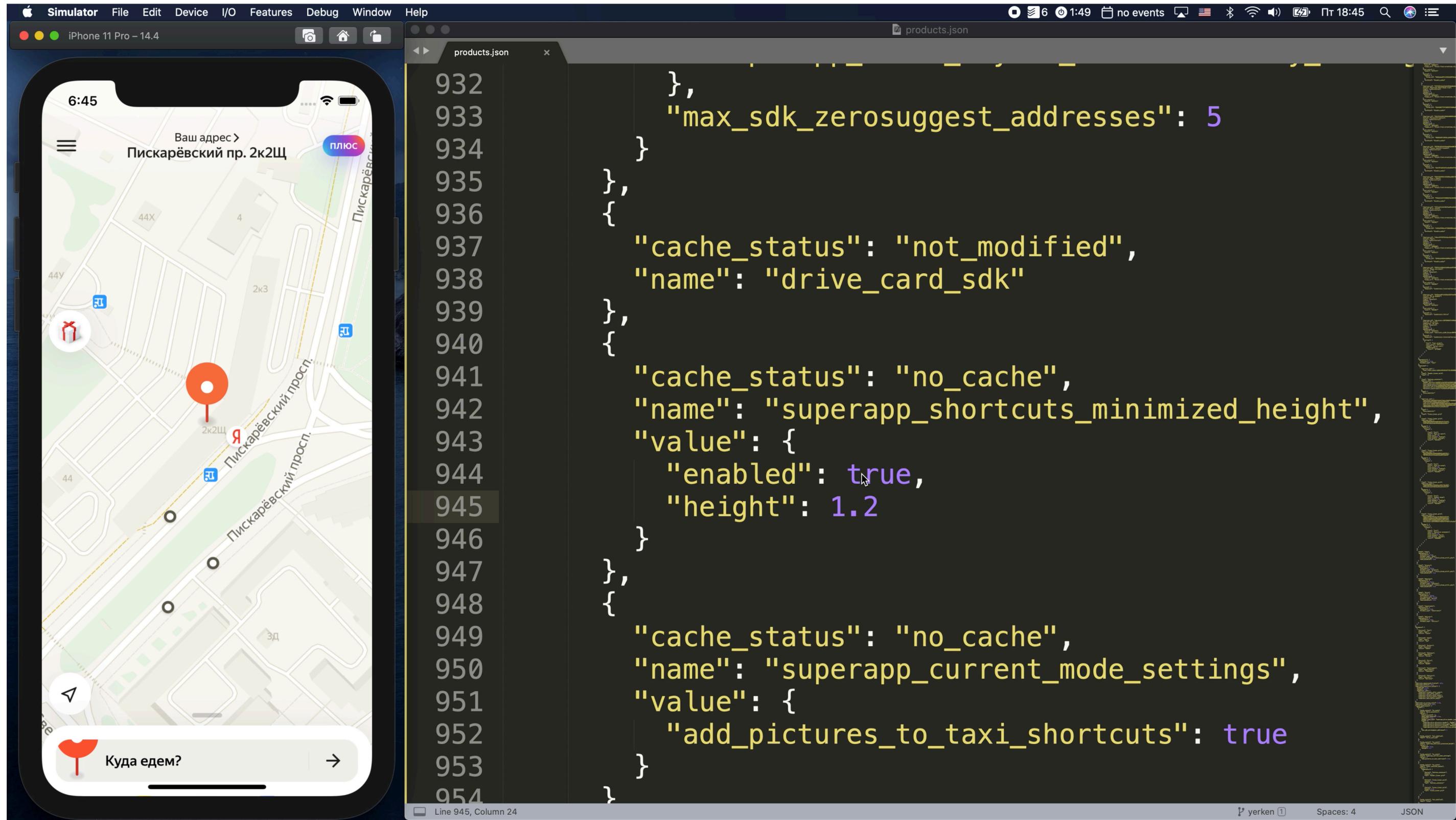
Оставляли только такси

Добавлялся нерелевантный шорткат

Увеличение размера блоков Еды, Лавки, Такси

Меняли внешний вид шапки

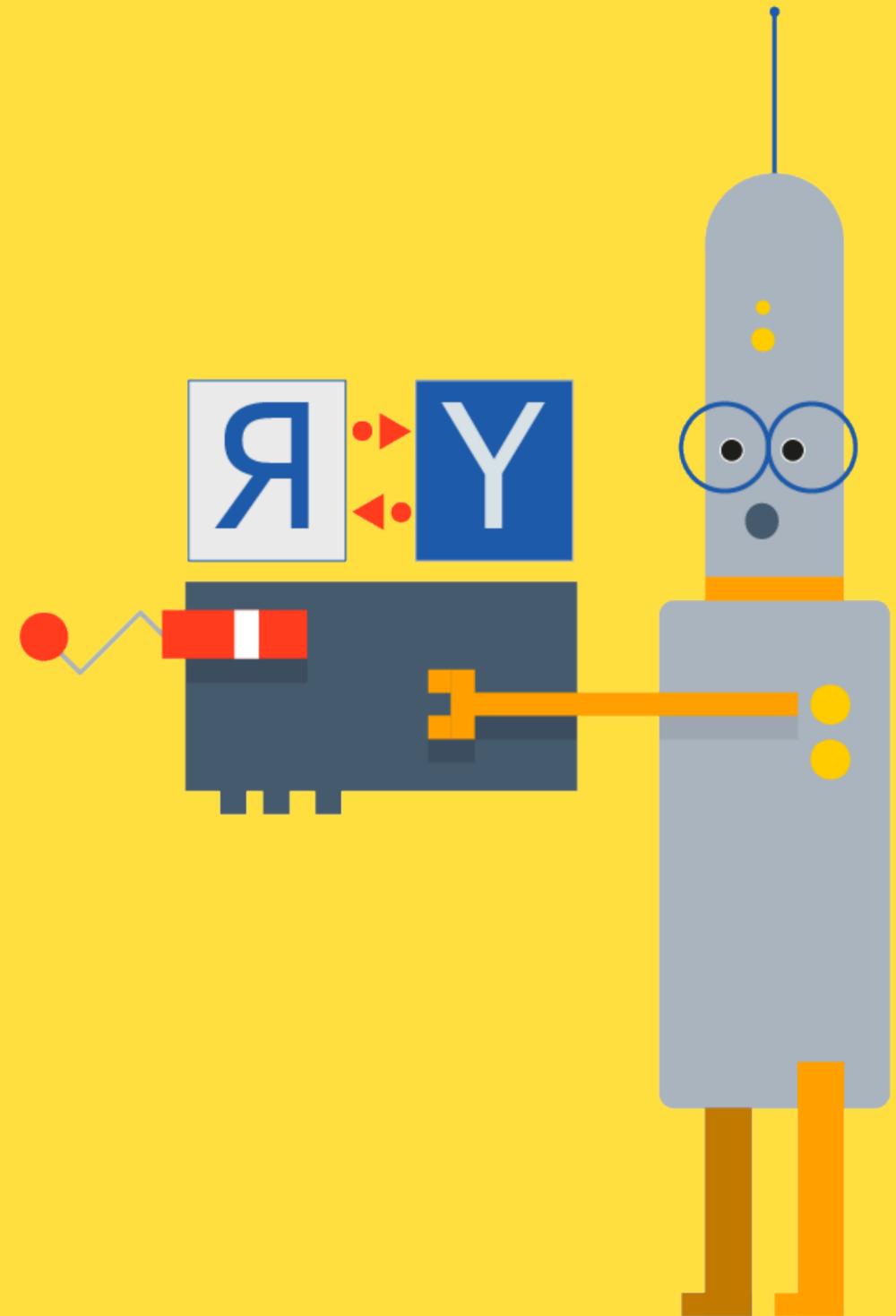
Эксперимент с высотой шторки



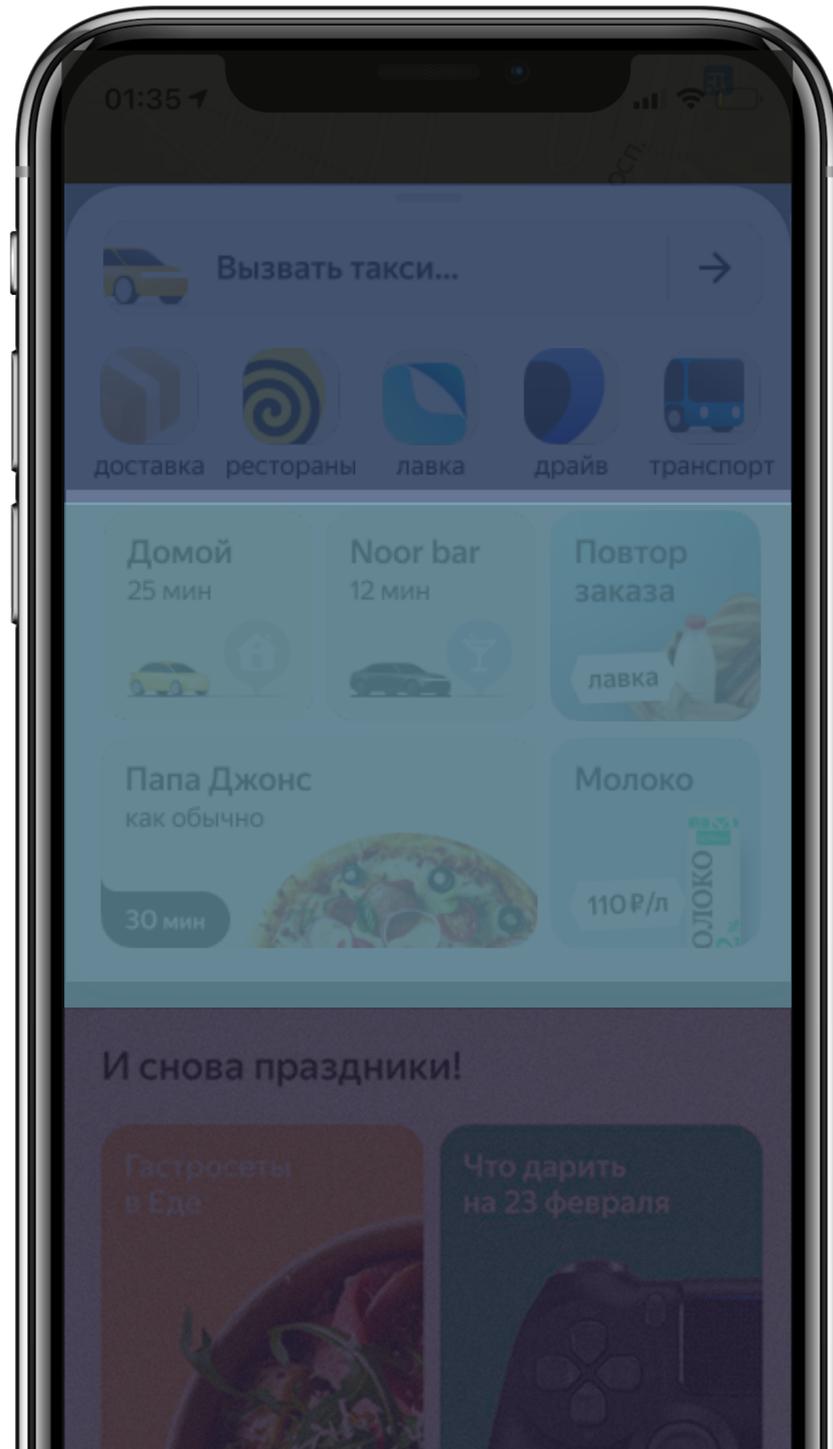
The image shows a side-by-side comparison of a mobile application interface and its configuration code. On the left, an iPhone 11 Pro simulator displays a map application with a search bar at the bottom containing the text "Куда едем?". On the right, a code editor shows a JSON file named "products.json" with the following configuration:

```
932 },  
933 "max_sdk_zerosuggest_addresses": 5  
934 }  
935 },  
936 {  
937 "cache_status": "not_modified",  
938 "name": "drive_card_sdk"  
939 },  
940 {  
941 "cache_status": "no_cache",  
942 "name": "superapp_shortcuts_minimized_height",  
943 "value": {  
944 "enabled": true,  
945 "height": 1.2  
946 }  
947 },  
948 {  
949 "cache_status": "no_cache",  
950 "name": "superapp_current_mode_settings",  
951 "value": {  
952 "add_pictures_to_taxi_shortcuts": true  
953 }  
954 }
```

**Возможность
подключения технологии
Machine Learning**



ML на шторке шорткатов

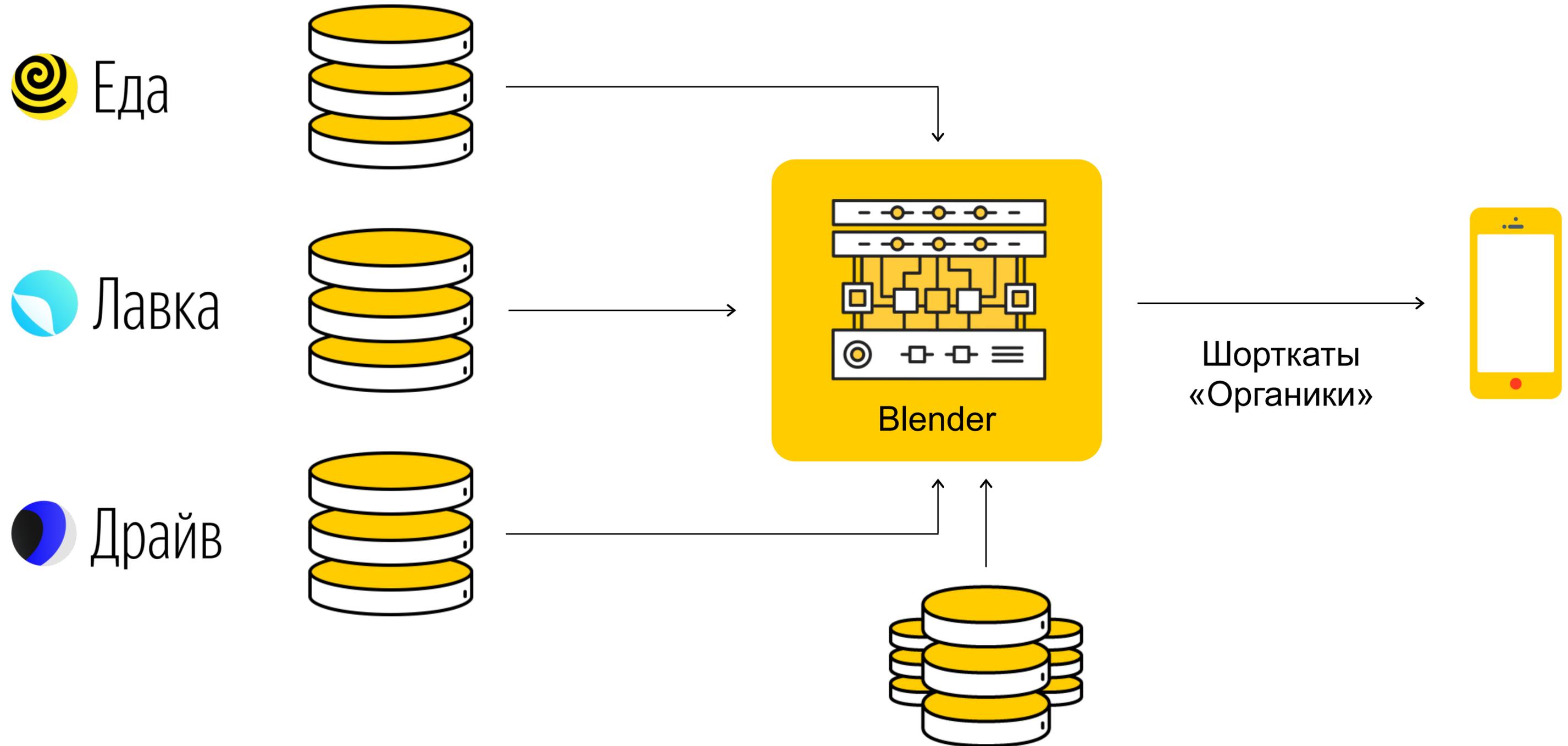


Статичная часть. Зависит от доступных сервисов в регионе

«Органика» - Часть шторки контролируемая при помощи ML. Выдает наиболее вероятные сценарии для пользователя

Медиа-контент

Как формируется «Органика»



План

Познакомились с Backend Driven UI

Разобрали Backend Driven UI на примере шрифтов

Возможности Backend Driven UI

Обратная совместимость

Итоги

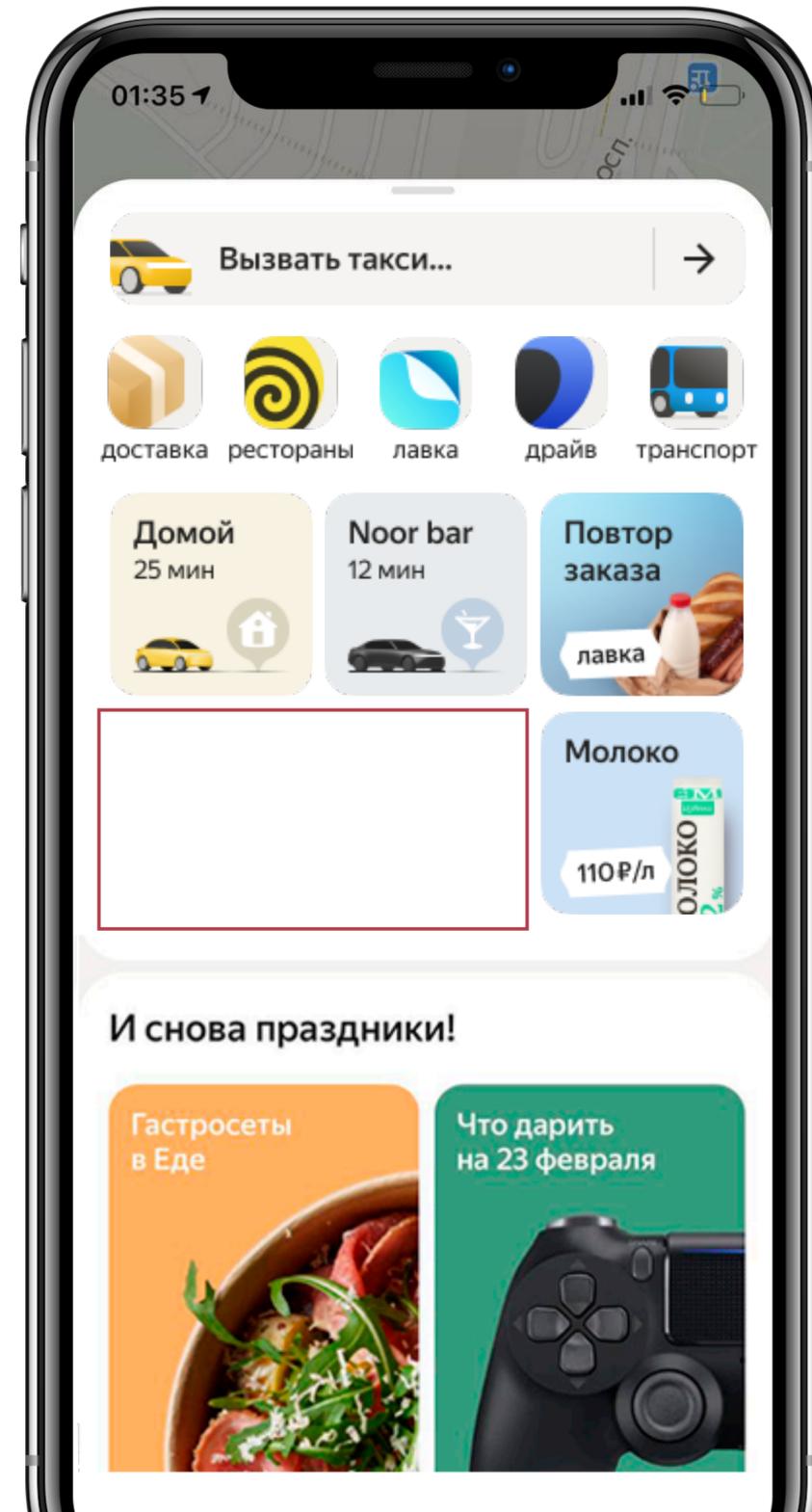
Старые клиенты не смогут отобразить новый компонент

Пострадают метрики. Будут ошибки в аналитике

ML будет плохо обучаться

Могут ухудшиться конверсии, из-за потери важного компонента

Можно поломать UI



Отправлять список поддерживаемых компонентов на backend



- | Backend должен готовить ответ, соответственно списку поддерживаемых компонентов на клиенте
- | Нужно следить за тем, чтобы список обновлялся при добавлении нового компонента

Новый Endpoint

Подойдет в случае, если структура ответа кардинально поменялась

Старые endpoints, всё равно придется поддерживать

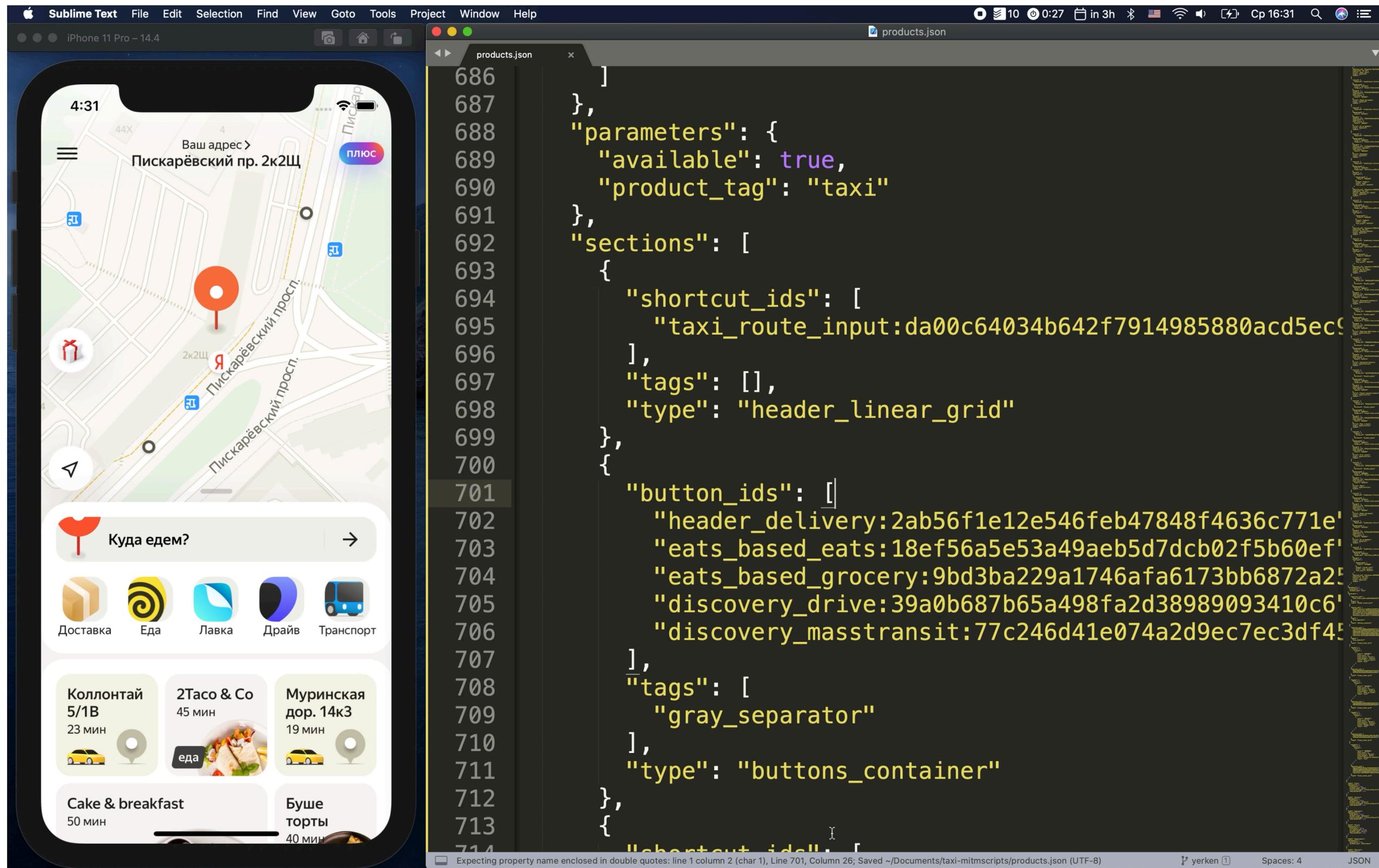
Что делать если клиент получил неизвестный компонент?

Подставлять дефолтную реализацию

Игнорировать неизвестный компонент

Игнорировать ответ целиком

Подставлять дефолтную реализацию



The image shows a side-by-side comparison between a mobile application interface and its underlying JSON configuration file. On the left, a simulated iPhone 11 Pro displays a map with a red location pin and a bottom navigation bar with icons for 'Доставка' (Delivery), 'Еда' (Food), 'Лавка' (Store), 'Драйв' (Drive), and 'Транспорт' (Transport). Below the navigation bar are several product cards, including 'Коллонтай 5/1В' (23 min), '2Тасо & Со' (45 min), 'Муринская дор. 14к3' (19 min), 'Cake & breakfast' (50 min), and 'Буше торты' (40 min). On the right, the 'products.json' file in a Sublime Text editor shows the JSON structure for a 'taxi' product. The JSON includes a 'parameters' object with 'available' set to true and 'product_tag' set to 'taxi'. It also contains a 'sections' array with a 'header_linear_grid' section and a 'buttons_container' section. The 'buttons_container' section lists several 'button_ids' with their corresponding product IDs and categories.

```
686 ]
687 },
688 "parameters": {
689   "available": true,
690   "product_tag": "taxi"
691 },
692 "sections": [
693   {
694     "shortcut_ids": [
695       "taxi_route_input:da00c64034b642f7914985880acd5ec9"
696     ],
697     "tags": [],
698     "type": "header_linear_grid"
699   },
700   {
701     "button_ids": [
702       "header_delivery:2ab56f1e12e546feb47848f4636c771e"
703       "eats_based_eats:18ef56a5e53a49aeb5d7dcb02f5b60ef"
704       "eats_based_grocery:9bd3ba229a1746afa6173bb6872a25"
705       "discovery_drive:39a0b687b65a498fa2d38989093410c6"
706       "discovery_masstransit:77c246d41e074a2d9ec7ec3df45"
707     ],
708     "tags": [
709       "gray_separator"
710     ],
711     "type": "buttons_container"
712   },
713   {
714     "shortcut_ids": [
```

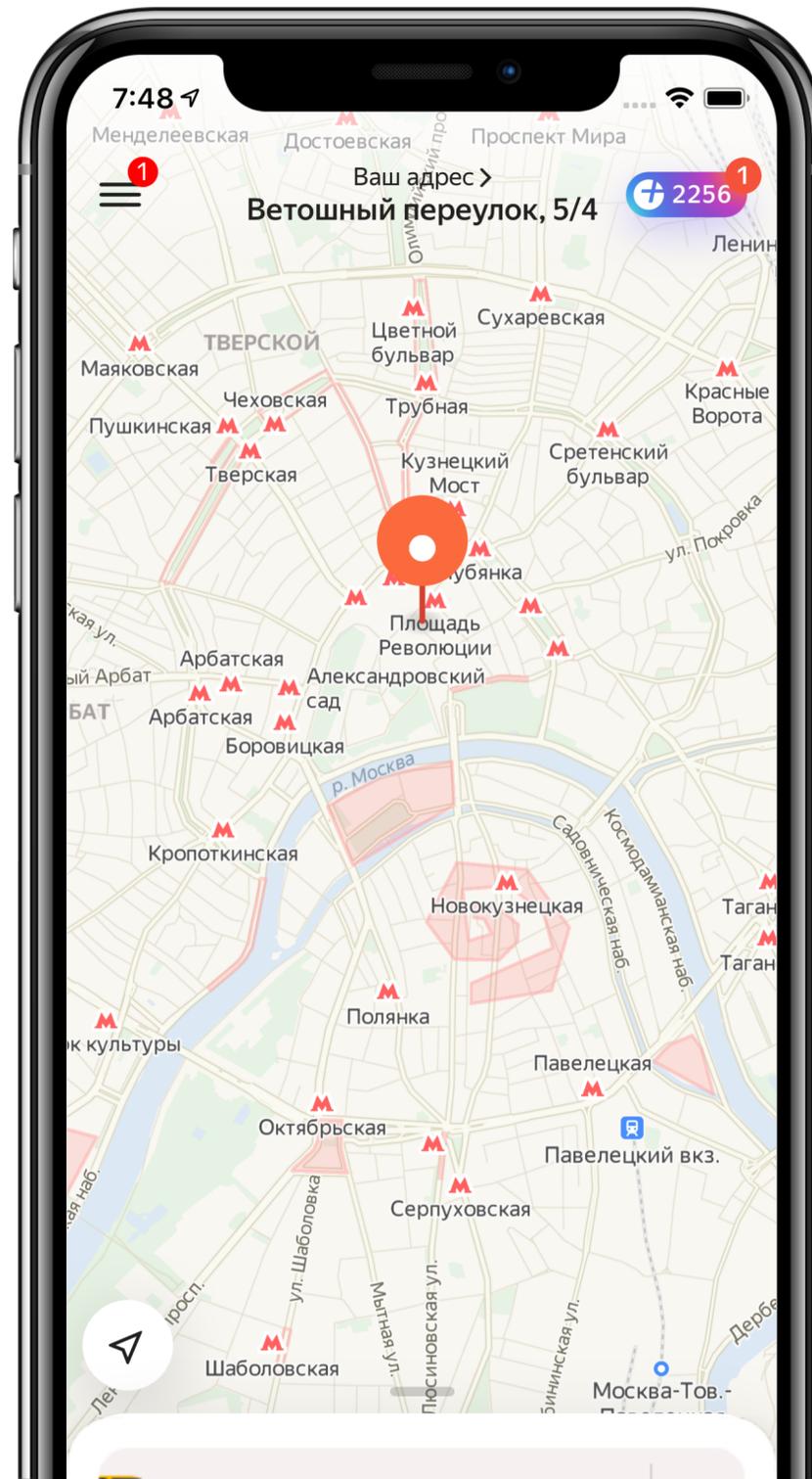
Игнорировать неизвестный компонент

The image shows a side-by-side comparison of a mobile application's UI and its underlying data structure. On the left, a simulated iPhone 11 Pro screen displays a map application. The map shows a location with a red pin and a search bar containing "Куда?". Below the map are icons for "Доставка" (Delivery), "Еда" (Food), "Лавка" (Store), "Драйв" (Drive), and "Транспорт" (Transport). At the bottom, there are several cards for nearby businesses, including "Коллонтай 5/1В" (23 min), "2Тасо & Со" (45 min), "Муринская дор. 14к3" (18 min), "Cake & breakfast" (50 min), and "Буше торты" (40 min).

On the right, a code editor window displays a JSON file named "products.json". The JSON data is a list of objects, with the visible portion corresponding to the "Коллонтай 5/1В" card. The JSON structure includes fields for "log" (a Yandex Maps URL), "position" (coordinates), "prefetch" ("route_eta"), "uri" (another Yandex Maps URL), "background" (color "#F5F2E4"), "height" (2), "width" (2), "overlays" (an array of overlay objects), "shortcut_id" ("7bbf5427d6f14396b1a04a5651687fe1"), "title" ("Коллонтай 5/1В"), "type" ("taxi:expected-destination"), and "action" (a deep link to a service).

```
229   "log": "ymapsbm1://geo?ll=30.453%2C59.912&spn=0.0",
230   "position": [
231     30.452888810846105,
232     59.91239636530226
233   ],
234   "prefetch": "route_eta",
235   "uri": "ymapsbm1://geo?ll=30.453%2C59.912&spn=0.0",
236 },
237 "background": {
238   "color": "#F5F2E4"
239 },
240 "height": 2,
241 "width": 2,
242 "overlays": [
243   {
244     "background": {
245       "color": "#C4BFA7"
246     },
247     "image_tag": "custom_shortcut_default_tag_2",
248     "shape": "poi"
249   }
250 ],
251 "shortcut_id": "7bbf5427d6f14396b1a04a5651687fe1",
252 "title": "Коллонтай 5/1В",
253 "type": "taxi:expected-destination"
254 },
255 },
256 "action": {
257   "deeplink": "vandex taxi: //external?service=eats&h"
```

Итоги



Гибкость

В дальнейшем сокращается Time To Market

Код будет чище: будет заметно меньше флажков и ветвлений в коде

Удобный инструмент для маркетинга

Реализация требует много времени

Сложность: увеличивается порог входа

Поддержка обратной совместимости



Спасибо

Еркебулан Абилдин

iOS Разработчик

 yerkebulan@yandex-team.ru

 [@yerkebulan](#)