Genfly  
Частное техническое задание на разработку

Стадия поиска предложений

Версия: 1.0b

Дата актуализации: 29.04.2019

[**1. Экран поиска**](#_22mlisevsqzd) **4**

[1.1. Структура экрана](#_didvuvxep5rq) 4

[Баннер](#_fzs4lv574w36) 4

[Иконка в навбаре](#_1cel2jfskuos) 4

[Табы переключения типа маршрута](#_7u3hf5r9rl2w) 4

[Поисковая форма.](#_vbwnpy3jr2ub) 4

[Тип маршрута “Roundtrip”](#_yohccow2pchp) 6

[Тип маршрута “One-way”](#_raxdkwna0usd) 6

[Тип маршрута “Multi-city”](#_9cyqy8yaxa7x) 7

[**1.2. Фильтры**](#_cb3e96lh3uji) **7**

[2.1. Фильтр Date](#_3fndibelnjxm) 7

[Элементы управления](#_xz7oah63d0sp) 8

[Календарь](#_340htl96wwxl) 8

[Интерактивные действия](#_gbbvvsp5atdd) 8

[Кнопка CONFIRM THE DATES](#_fhecwy5o2eta) 9

[Условия отображения](#_tkz5n31h2xw) 9

[Интерактивные действия](#_oh0enblu15xq) 9

[2.2. Фильтр Time](#_vhn66hekn44u) 9

[Элементы управления](#_w5bti8ths6ky) 10

[Набор чекбоксов DEPARTURE/ARRIVAL](#_d1fmjj3njuhn) 10

[Интерактивные действия](#_1c0svqzdiwdx) 10

[Значение по умолчанию](#_kkb7d5n7tl7s) 10

[2.3. Фильтр Passengers](#_gbcb5s4nbv9n) 10

[2.4. Фильтр Baggage](#_3gwtvzqsx2k4) 11

[Элементы управления](#_fcju9uipi0mk) 12

[2.5. Фильтр Cabin](#_6e0nldjk53w1) 12

[Элементы управления](#_jifaokbi8t2f) 12

[2.6. Фильтр Stops](#_sc8uixl19cv9) 13

[Элементы управления](#_7gbq469kw1go) 13

[Переключатель STOPS](#_akml4okz76j7) 13

[Состояния](#_orkzvs37skr7) 13

[1.3. Кнопка SEND ME RESULTS](#_lyl0n6yjj1o0) 13

[**2. Auto-complete work**](#_slvorl8uia5o) **13**

[**3. Calendar**](#_5tnzt419pyfo) **13**

[**4. Спецификация поискового запроса**](#_5lvnqai11nub) **13**

[4.1. Примеры поисковых запросов](#_7x9t3zf27hgk) 15

[Прямой рейс](#_tuund6774dyh) 15

[Roundtrip](#_z87ym8byj27t) 16

[4.2. Поведение в момент ожидания ответа от back-end](#_pc35ansy7kld) 16

[4.3. Интерпретация ответа бэкенда на запрос](#_hnmh1ca9sukw) 16

[Успешный ответ](#_oltppv759gtp) 16

[Возвращаемый результат](#_4obqxnyfd06s) 17

[Ошибка, нет результатов](#_5cc3qp9v2i43) 17

[Возможные действия пользователя](#_6r9jzvrd3aq6) 17

[Схема ответа](#_mwvtk1gegonz) 18

[Ошибка, таймаут](#_pec9vxjb29pt) 19

[Возможные действия пользователя](#_bd75tbes9s5f) 19

[**5. Справочники**](#_hz8gjm4lf6mr) **19**

[5.1. Cabin](#_gbf2qd3mm9qf) 19

[5.2. Stops](#_6du990z2dnbz) 19

[**6. Комплитер**](#_wg41n1rix8mk) **20**

[6.1. Схема обновления комплитера](#_rv6mxt7oq9z5) 20

# 1. Экран поиска

## 1.1. Структура экрана

### Баннер

Баннер имеет пропорции ширины и высоты – 3:2. Например, для iPhone 8 размер баннера 375x250px, а для iPhone SE – 320x213px.

Содержимое баннера.

### Иконка в навбаре

Переход в профиль пользователя

### Табы переключения типа маршрута

* Roundtrip
* One-way
* Multi-city

Переключаться между типами маршрута можно всегда. При этом данные в форме должны сохраниться.

Например, при выбранном Roundtrip был указан маршрут Нью-Йорк – Амстердам, 13–20 апреля.

Если переключиться на One-way, маршрут должен быть Нью-Йорк – Амстердам, 13 апреля.

А если переключиться на Multi-way, то в первом сегменте должен отобразиться маршрут Нью-Йорк – Амстердам, 13 апреля.

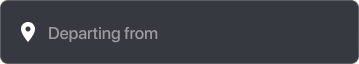
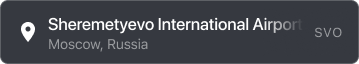
### Поисковая форма.

Форма состоит из двух групп полей:

1. Маршрут  
   Набор полей зависит от выбранного типа маршрута и будет описан ниже.
2. Фильтры  
   Фиксированный набор полей для всех типов маршрутов:
   * Passengers (по умолчанию – 1 Adult)
   * Cabin (по умолчанию – Economy)
   * Stops (по умолчанию – Any)
   * Baggage (по умолчанию – Any)

Каждый из фильтров будет описан ниже.

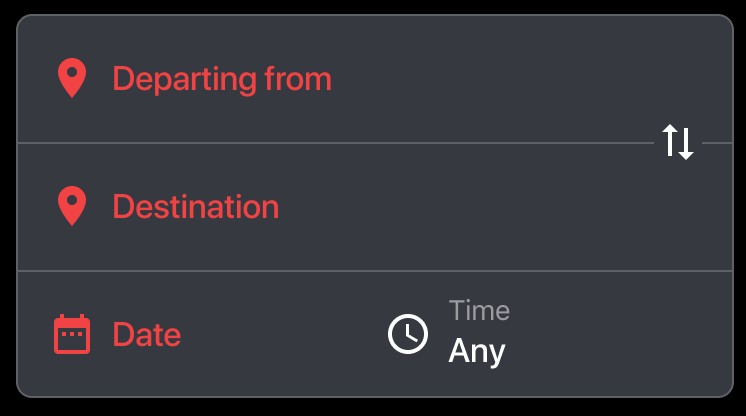
Каждое поле может быть представлено тремя состояниями:

* Незаполненное  
  
* Заполненное  
  
* Ошибка  
  

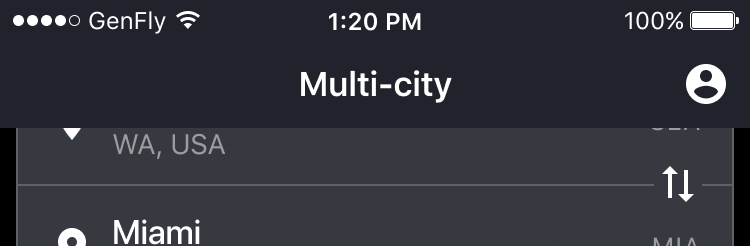
Кнопка отправки формы “Send me results”.

Кнопка всегда зафиксирована внизу экрана и доступна для нажатия.

В случае, если некоторые поля не заполнены, и пользователь нажимает на кнопку – в незаполненных полях иконка и лейбл становятся красными, подсвечивая ошибку:



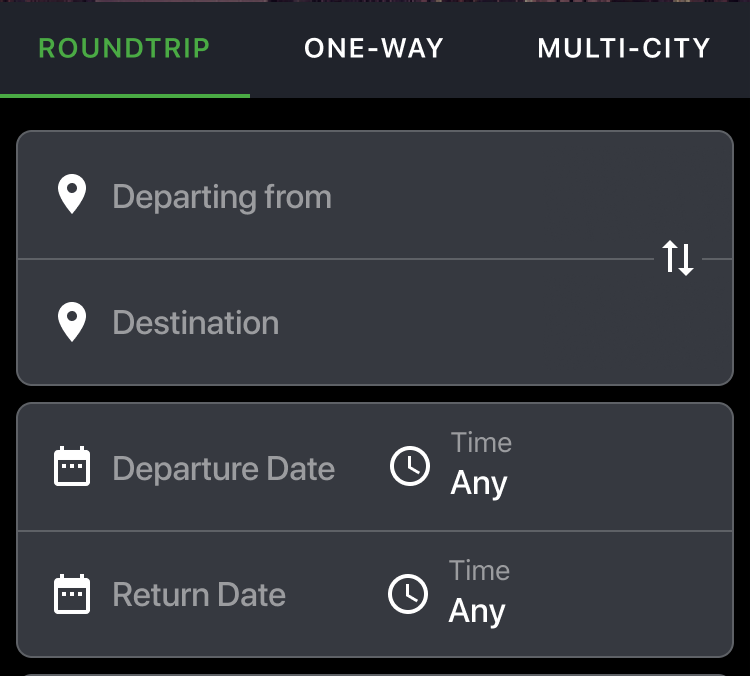
Navbar



При скроле экрана должен появляться Navbar, в заголовке которого должен быть выбранный тип маршрута.

Как пример похожего поведения, можно привести экран исполнителя в Apple Music, где имя исполнителя изначально указано в нижней части верхнего “баннера” и нет Navbar’a, а при сколле Navbar появляется и имя исполнителя переходит в него. Только в нашем случае никакой сложной анимации при сколле вниз делать не нужно.

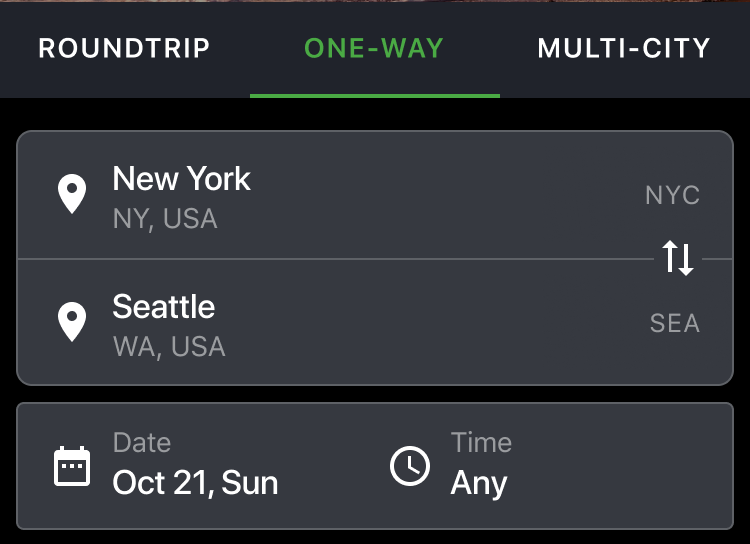
#### Тип маршрута “Roundtrip”



#### Элементы управления

* Departing from - поле выбора аэропорта вылета. Поле является обязательным для выбора;
* Destination - поле выбора аэропорта прилёта. Поле является обязательным для выбора;
* Departure Date #1. Дата первого вылета. Поле является обязательным для выбора;
* Time #1. Желательное время первого вылета. Поле явлется опциональным для выбора; значение по умолчанию – Any;
* Return Date #2. Дата обратного вылета. Поле является обязательным для выбора
* Time #2. Желательное время обратного вылета. Поле явлется опциональным для выбора; значение по умолчанию – Any;
* Иконка с двойной вертикальной стрелкой. При нажатии на иконку значения полей Departing from и Destination меняются местами. Это можно делать, даже если одно из полей пустое. Например, если указан маршрут Нью-Йорк – Пусто, то после нажатия на иконку Поменять местами будет Пусто – Нью-Йорк.

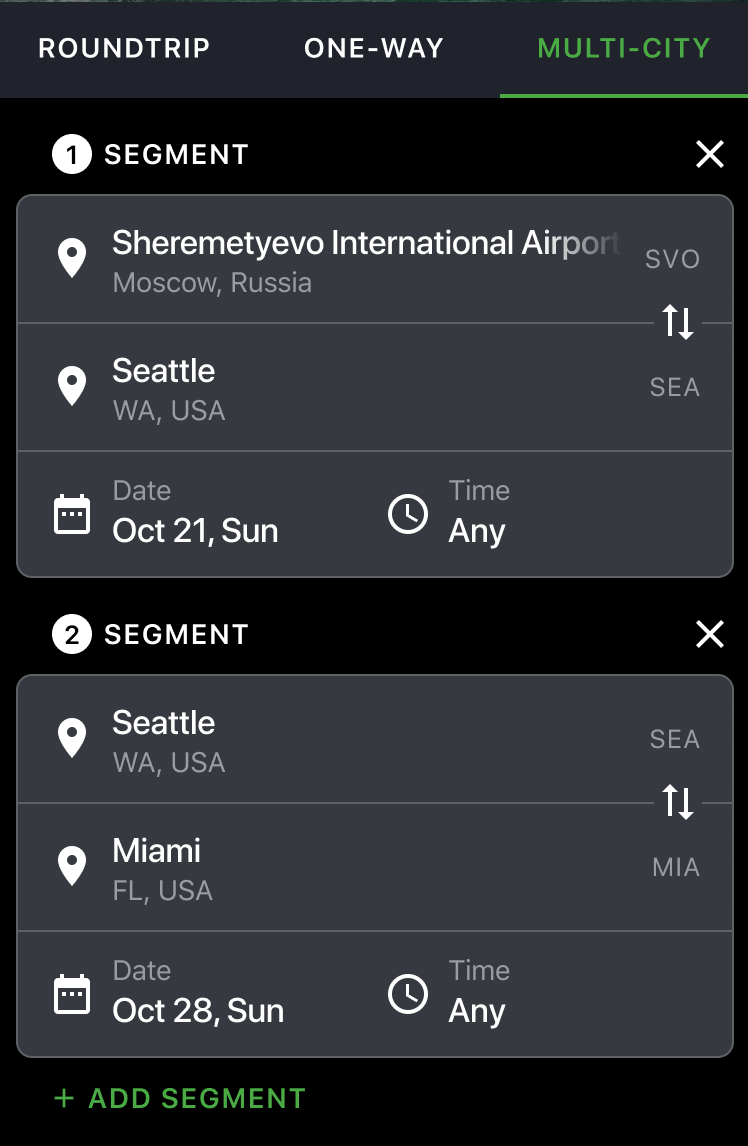
#### Тип маршрута “One-way”



#### Элементы управления

* Departing from - поле выбора аэропорта вылета. Поле является обязательным для выбора;
* Destination - поле выбора аэропорта прилёта. Поле является обязательным для выбора;
* Date. Дата первого вылета. Поле является обязательным для выбора;
* Time. Желательное время первого вылета. Поле явлется опциональным для выбора; значение по умолчанию – Any;
* Иконка с двойной вертикальной стрелкой. При нажатии на иконку значения полей Departing from и Destination меняются местами. Это можно делать, даже если одно из полей пустое. Например, если указан маршрут Нью-Йорк – Пусто, то после нажатия на иконку Поменять местами будет Пусто – Нью-Йорк.

#### Тип маршрута “Multi-city”



#### Элементы управления

* Departing from - поле выбора аэропорта вылета. Поле является обязательным для выбора;
* Destination - поле выбора аэропорта прилёта. Поле является обязательным для выбора;
* Date. Дата первого вылета. Поле является обязательным для выбора;
* Time. Желательное время первого вылета. Поле явлется опциональным для выбора; значение по умолчанию – Any;
* Иконка с двойной вертикальной стрелкой. При нажатии на иконку значения полей Departing from и Destination меняются местами. Это можно делать, даже если одно из полей пустое. Например, если указан маршрут Нью-Йорк – Пусто, то после нажатия на иконку Поменять местами будет Пусто – Нью-Йорк.
* Кнопка “+ADD SEGMENT”. При добавлении нового сегмента значение поля Departing from в нем должно быть таким же, что указано в Destination предыдущего сегмента. Поля Date и Time у нового сегмента всегда пустые.
* Кнопка “X” напротив заголовка сегмента. При нажатии происходит удаление соответствующего сегмента.

##### Ограничения

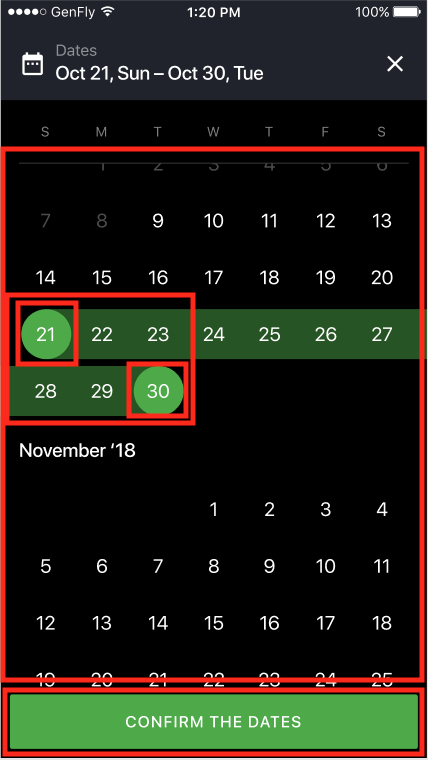
* Минимальное кол-во сегментов – 2;
* Максимальное кол-во сегментов – 6;
* Все сегменты пронумерованы, менять порядок сегментов пользователь не может.

## 1.2. Фильтры

### 2.1. Фильтр Date

Выбор даты вылета и возвращения (для обратных вылетов).

Общий вид:



*Примечание: красным цветом выделены элементы управления (см.описание ниже).*

#### Элементы управления

##### Календарь

Календарь отображает даты с помесячной группировкой. Дефолтный вид - первая отображаемая строка содержит текущую дату.

###### Интерактивные действия

* Смахивание пальцем вверх/вниз по области календаря прокручивает его вперед/назад; Календарь “назад” доступен на минус 1 месяц от текущей даты;
* Нажатие на конкретной дате в случае перелёта one-way или multi-city выбирает дату вылета для рейса (или для текущего сегмента в случае multi-city);
* (если до этого ничего не было выбрано) Нажатие на конкретной дате в случае перелёта туда-обратно выбирает дату прямого вылета. Выбранная дата фиксируется и отображается в строке Dates;
* (если до этого уже была выбрана дата прямого вылета) Нажатие на конкретной дате в случае перелёта туда-обратно выбирает дату обратного вылета. При этом даты прямого и обратного вылетов сортируются по возрастанию, чтобы исключить ситуацию, в которой дата прямого вылета наступает позднее, чем дата обратного. Дни между выбранными датами отображаются в виде полупрозрачной полосы. Выбранная дата фиксируется и отображается в строке Dates;
* (если до этого уже была выбрана дата обратного вылета) Нажатие на конкретной дате в случае перелёта туда-обратно выбирает дату прямого вылета. Выбранные до этого даты сбрасываются. Выбранная новая дата фиксируется и отображается в строке Dates;
* Нажатие на конкретной дате с протяжкой (применимо только для перелёта туда-обратно) выбирает дату вылета (дата, где было нажатие) и обратного вылета (дата, где завершилась протяжка). При этом даты прямого и обратного вылетов сортируются по возрастанию, чтобы исключить ситуацию, в которой дата прямого вылета наступает позднее, чем дата обратного. Если до этого уже была выбрана какая-либо дата (либо даты), они заменяются на новые. Дни между выбранными датами отображаются в виде полупрозрачной полосы. Выбранные даты фиксируются и отображаются в строке Dates;
* Нажатие на конкретной дате с протяжкой вниз/вверх за область календаря (применимо только для перелёта туда-обратно) выбирает дату прямого вылета (дата, где было нажатие) и обратного вылета (дата, где завершилась протяжка). При этом выполняется прокрутка календаря до тех пор, пока пользователь не вернется в область календаря либо не отпустит палец. При этом даты прямого и обратного вылетов сортируются по возрастанию, чтобы исключить ситуацию, в которой дата прямого вылета наступает позднее, чем дата обратного. Если до этого уже была выбрана какая-либо дата (либо даты), они заменяются на новые. Дни между выбранными датами отображаются в виде полупрозрачной полосы. Выбранные даты фиксируются и отображаются в строке Dates.

##### Кнопка CONFIRM THE DATES

###### Условия отображения

* Если не выбрана дата первого вылета, кнопка отображается как неактивная;
* (для перелётов туда-обратно) Если не выбрана дата обратного вылета, кнопка отображается как disabled;

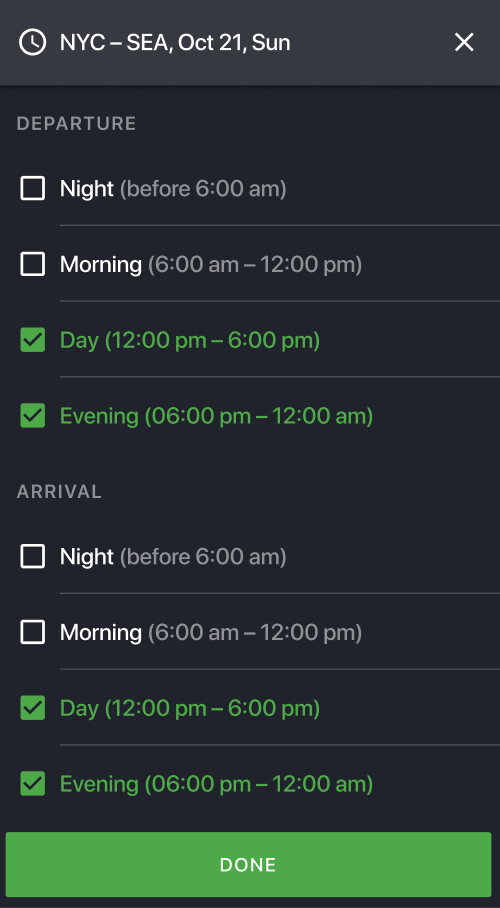
###### Интерактивные действия

* При нажатии сохраняется выбранная дата прямого (для всех перелётов) и обратного вылетов (для перелётов туда-обратно); окно выбора даты закрывается.

### 2.2. Фильтр Time

Фильтр перелётов по времени вылета/прилёта.

Общий вид:



#### Элементы управления

##### Набор чекбоксов DEPARTURE/ARRIVAL

Задают параметры фильтра для времени вылета/прилёта рейсов.

###### Интерактивные действия

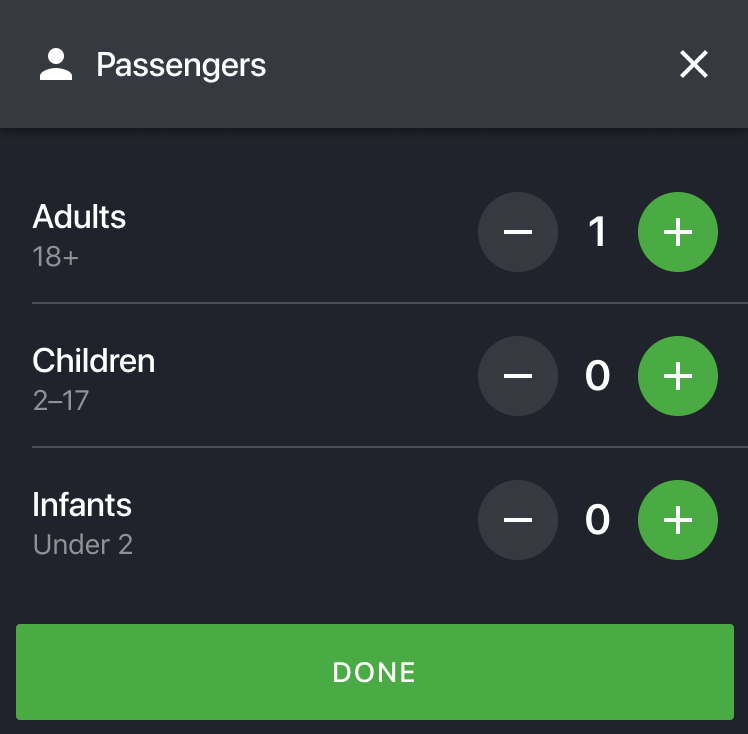
Включение чекбоса добавляет связанный с ним диапазон времени в фильтр отображаемых результатов

###### Значение по умолчанию

По умолчанию включены все чекбоксы / приемлемо любое время вылета/прилёта.

### 2.3. Фильтр Passengers

Дефолтное значение фильтра (=выбран один взрослый пассажир):



(+) добавляет одного пассажира к выбранной категории

(-) убирает одного пассажира из выбранной категории

При этом:

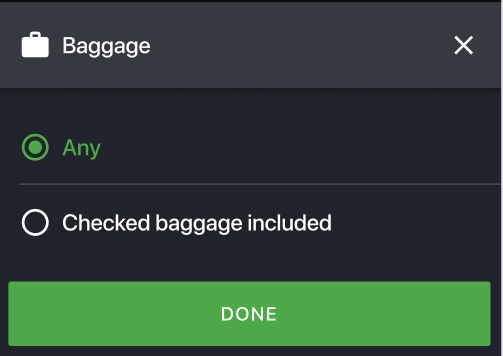
1. Если Adults указано 1 (дефолтно), то (-) не активен (меньше одного взрослого пассажира в запросе указать нельзя)
2. Количество Infant должно быть не больше количества Adults. Например, если указан 2 Adults, то максимально допустимое количество Infants – 2. При указании максимально допустимого количества infants, кнопка (+) у infants должна быть не активна.
3. Если указано несколько adults и такое же количество infants, то кнопка (-) у adults должна быть не доступна. Например, указано 3 adults + 3 infants, удалить одного adult нельзя, тк получим невалидную ситуацию.

Общее количество выбранных пассажиров должно быть не более 8. Например, 4 adults + 3 children + 1 infant. Или 7 adults + 1 child. Если количество выбранных пассажиров равно 8, то во всех категориях пассажиров (+) должен быть не доступен.

### 2.4. Фильтр Baggage

Фильтрация тарифов по наличию бесплатного багажа, включенного в тариф.

Общий вид фильтра:



#### Элементы управления

Тип: переключатель (radio)

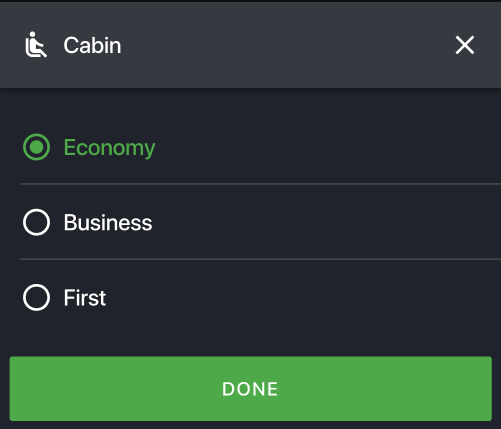
Состояния:

* Any *(по умолчанию)*  
  Выполнять поиск по всем тарифам, независимо от условий включения бесплатного багажа;
* Checked baggage included  
  Выполнять поиск только по тарифам с включенным бесплатным багажом

### 2.5. Фильтр Cabin

Выбор класса бронирования.

Общий вид фильтра:



#### Элементы управления

Тип: переключатель (radio)

Состояния:

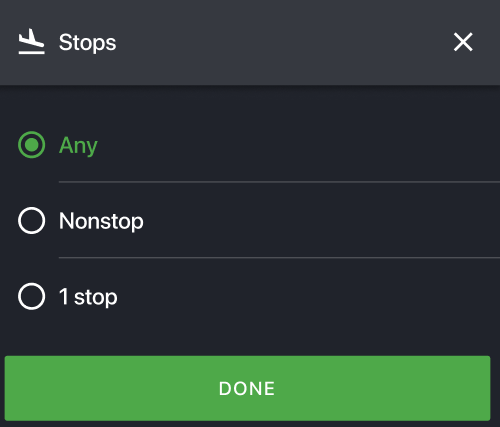
* Economy  
  Выполнять поиск по тарифам экономического класса;
* Business  
  Выполнять поиск по тарифам бизнес-класса;
* First  
  Выполнять поиск по тарифам первого класса

По умолчанию: Economy

### 2.6. Фильтр Stops

Выбор перелётов по максимальному количеству стыковок.

Общий вид:



#### Элементы управления

##### Переключатель STOPS

###### Состояния

* Any *(по умолчанию)*  
  Возвращать все варианты перелётов независимо от количества стыковок.
* Nonstop  
  Возвращать только прямые рейсы.
* 1 stop  
  Возвращать перелёты с количеством стыковок НЕ БОЛЕЕ 1 (прямые + 1 стыковка)

## 1.3. Кнопка SEND ME RESULTS

Отправляет запрос на поиск и отправку результатов поиска пользователю.

# 2. Auto-complete work

# 3. Calendar

# 4. Спецификация поискового запроса

{

"required": [

"baggage",

"cabin",

"passengers",

"segments",

"stops"

],

"$schema": "http://json-schema.org/draft-04/schema#",

"type": "object",

"properties": {

"cabin": {

"maxLength": 1,

"pattern": "^[FCYWMB]$",

"minLength": 1,

"type": "string"

},

"stops": {

"type": "string",

"enum": [

"any",

"nonstop",

"at\_most\_one"

]

},

"passengers": {

"required": [

"adult"

],

"type": "object",

"properties": {

"child": {

"oneOf": [

{

"type": "null"

},

{

"type": "number"

}

]

},

"infant": {

"oneOf": [

{

"type": "null"

},

{

"type": "number"

}

]

},

"adult": {

"type": "number"

}

}

},

"segments": {

"items": {

"required": [

"arrival\_time",

"date",

"departure\_time",

"from",

"to"

],

"type": "object",

"properties": {

"to": {

"pattern": "^[A-Z0-9]{3}$",

"type": "string"

},

"from": {

"pattern": "^[A-Z0-9]{3}$",

"type": "string"

},

"date": {

"pattern": "[1-9][0-9]{3}-(0[1-9]|1[0-2])-([0-2][0-9]|3[0-1])",

"type": "string"

},

"departure\_time": {

"type": "string",

"enum": [

"any",

"night",

"morning",

"day",

"evening"

]

},

"arrival\_time": {

"type": "string",

"enum": [

"any",

"night",

"morning",

"day",

"evening"

]

}

}

},

"type": "array"

},

"baggage": {

"type": "string",

"enum": [

"any",

"included"

]

}

}

}

## 4.1. Примеры поисковых запросов

1 взрослый, из Екатеринбурга в Стамбул (время и прибытие любое), эконом

### Прямой рейс

{

"segments": [

{

"from": "SVX",

"departure\_time": "any",

"to": "IST",

"arrival\_time": "any",

"date": "2019-05-23"

}

],

"passengers": {

"adult": 1,

"child": null,

"infant": null

},

"cabin": "Y",

"stops": “any”,

“baggage”: “any”

}

### Roundtrip

{

"segments": [

{

"from": "SVX",

"departure\_time": "any",

"to": "IST",

"arrival\_time": "any",

"date": "2019-05-23"

},

{

"from": "IST",

"departure\_time": "any",

"to": "SVX",

"arrival\_time": "any",

"date": "2019-05-25"

}

],

"passengers": {

"adult": 1,

"child": null,

"infant": null

},

"cabin": "Y",

"stops": “any”,

“baggage”: “any”

}

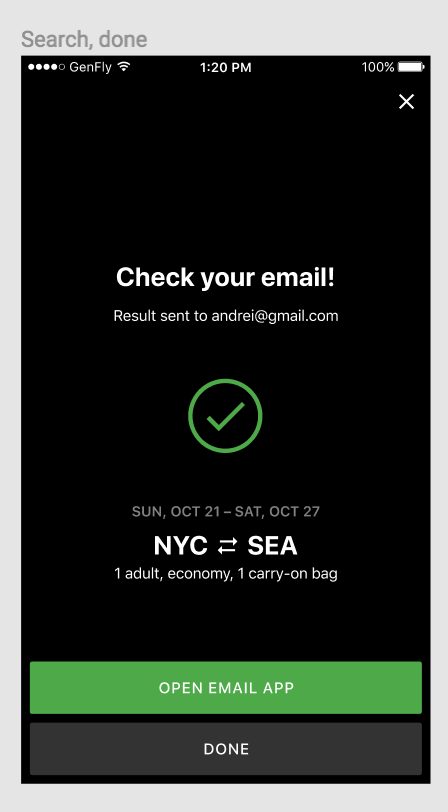
## 4.2. Поведение в момент ожидания ответа от back-end

Timeout (60 сек или меньше? ) // 500 – отсутствие ответа.

## 4.3. Интерпретация ответа бэкенда на запрос

### Успешный ответ

При успешном ответе приложение должно продолжить работу по основному сценарию, то есть перейти на “Search, done”.



#### Возвращаемый результат

{“status”: “ok”}

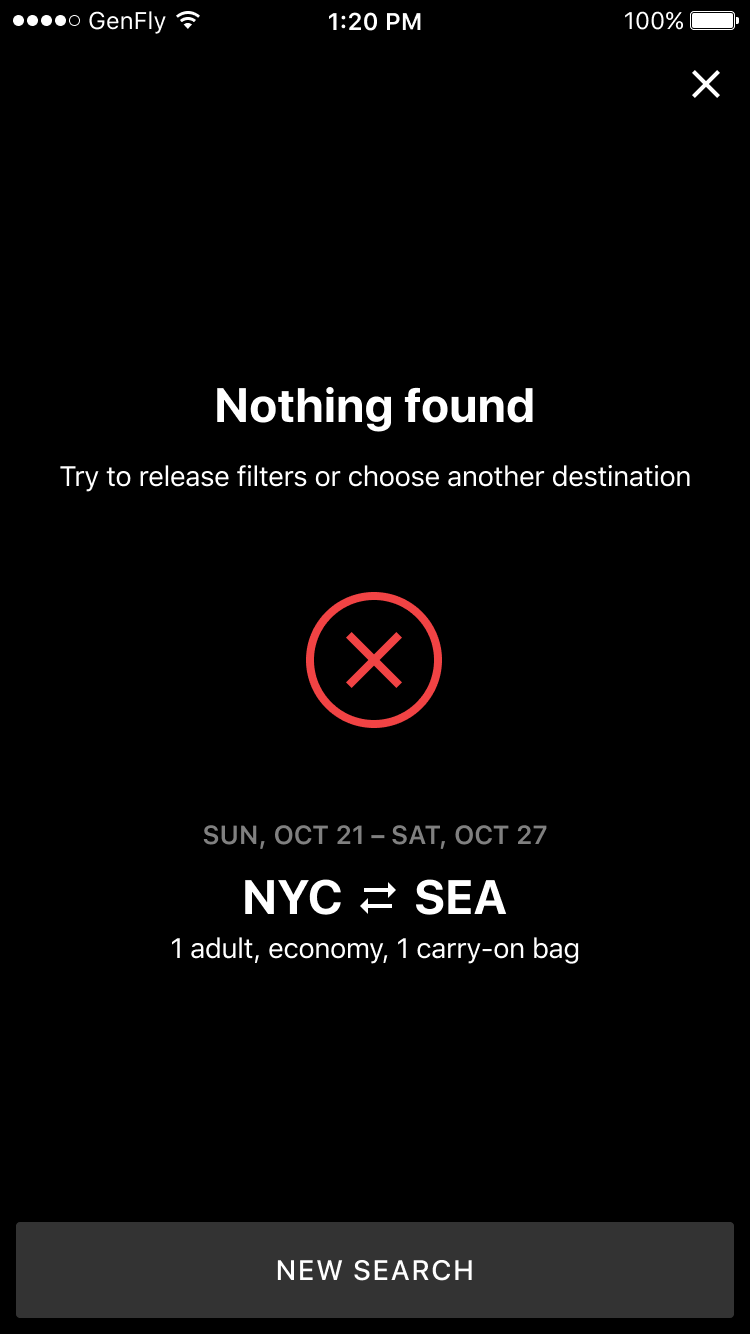
### Ошибка, нет результатов

Если сервер вернул ошибку, то необходимо отобразить экран с “Something went wrong”. Означает, что мы не нашли подходящих вариантов.

#### Возможные действия пользователя

Сделать запрос на новый поиск

Приложение должно показать пользователю скрин Search, nothing found:



#### Схема ответа

{

"anyOf": [

{

"required": [

"status"

],

"type": "object",

"properties": {

"status": {

"type": "string",

"enum": [

"empty\_result"

]

}

}

},

{

"required": [

"status"

],

"type": "object",

"properties": {

"status": {

"type": "string",

"enum": [

"ok"

]

}

}

}

],

"$schema": "http://json-schema.org/draft-04/schema#"

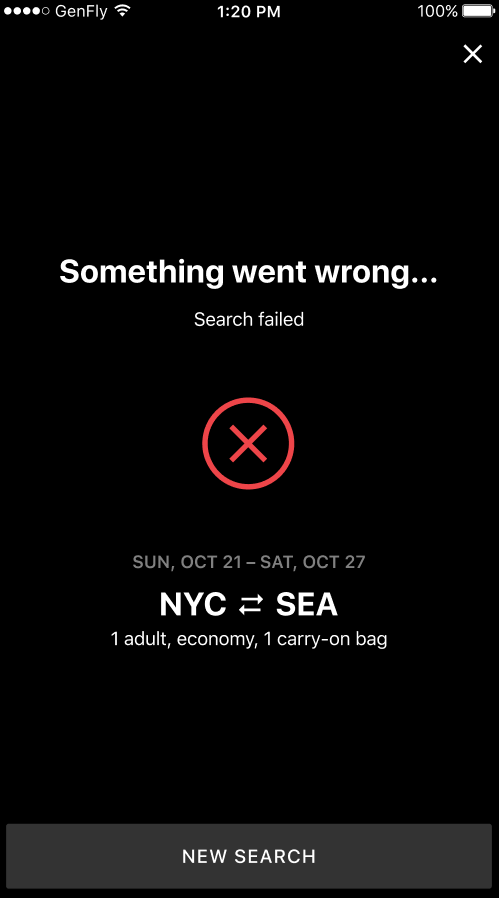
}

### Ошибка, таймаут

В случае неполучения ответа от сервера в течение установленного таймаута (60 сек.), возвращать ошибку 500 (отсутствие ответа). Пользователь получает стандартное сообщение об ошибке (см.скриншот “Search, failed”).

#### Возможные действия пользователя

Сделать запрос на новый поиск



# 5. Справочники

## 5.1. Cabin

* Economy - “Y”
* Business - “C”
* First - “F”

## 5.2. Stops

* Any - “any”
* No stops - “nonstop”
* 1 stop - “at\_most\_one”

# 6. Комплитер

Запрос по адресу `api/completer` возвращает геоданные (города, аэропорты) в оптимальной для поиска форме [сжатого префиксного дерева] (<https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B5%D1%84%D0%B8%D0%BA%D1%81%D0%BD%D0%BE%D0%B5_%D0%B4%D0%B5%D1%80%D0%B5%D0%B2%D0%BE>).

Каждый узел дерева состоит из префикса (prefix), списка дочерних узлов (children) и информации об узле (item), если узлу соответствует объект геоданных.

Пример дерева:

("",null)

-- ("London", ...)

-- ("San", null)

-- ("-Diego", ...)

-- ("kt-Petersburg", ...)

-- ("A",...)

-- ("lmaty", ...)

-- ("d", ...)

-- ("dis Ababa", ...)

-- ("elaida", ...)

Имя объекта составляется из префиксов в пути к объекту. Листья дерева всегда соответствуют какому-то объекту. Но объекты, в принципе, могут быть не только в листьях дерева.

В ответе передается информация об аэропортах и городах.

Если городу соответствует один аэропорт и коды IATA города и аэропорта совпадают, то передается только информация о городе.

Поля в объекте обозначают следующее:

* name - имя объекта (соответствует префиксам в пути к объекту)
* iata - IATA-код объекта
* type - airport / city
* lng - Longitude (долгота)
* lat - Latitude (широта)
* hint - вспомогательный текст для различения объектов с одинаковыми именами
* importance - важность (используется для сортировки).

Более употребимый объект имеет большее значение importance.

Чтобы получить список вариантов по начальным буквам надо выполнить следующие действия:

1. Найти поддерево, соответствующее начальным буквам
2. Собрать все узлы поддерева, у которых в поле item не null, определить их имена
3. Отсортировать полученные узлы по убыванию значения в поле item.importance

## 6.1. Схема обновления комплитера

При открытии экрана с поиском необходимо сделать запрос, чтобы получить данные Completer для ввода геоданных (аэропорт, город). Построить внутреннее представление, чтобы полученные данные можно использовать для поиска направлений.

Если поисковый запрос вернул ошибку, связанную с полями “from” или “to”, то это значит, что данные на клиенте устарели и нужно снова сделать запрос к комплитеру, чтобы перестроить локальный индекс.