

## Техническое задание на разработку мобильного приложения

[Введение](#)

[Макет](#)

[Аппаратные и программные требования](#)

[Требования к коду приложения](#)

[Работа с API](#)

[Основные сведения](#)

[Примеры запросов](#)

[Пример запроса авторизации пользователя](#)

[Пример запроса информации о текущем пользователе](#)

[Ответ сервера](#)

[Batch-запросы](#)

[Авторизация пользователя](#)

[Разлогинивание пользователя](#)

[Регистрация устройства для push-уведомлений](#)

[Авторизация через сторонние OAuth сервисы](#)

[Список городов](#)

[Тutorial](#)

[Главный экран](#)

[Форма подачи вопроса](#)

[Первый экран. Текст вопроса](#)

[Валидация](#)

[Приложить фото](#)

[Второй экран. О вас](#)

[Экран определения города](#)

[Валидация](#)

[Третий экран. Тип вопроса](#)

[Платный \(стоит по умолчанию\)](#)

[Бесплатный](#)

[Публикация бесплатного вопроса](#)

[Оплата вопроса](#)

[Последовательность API вызовов для публикации вопроса](#)

[Экран успеха](#)

[Для авторизованного пользователя](#)

[Черновики](#)

[Мои вопросы](#)

[Страница вопроса](#)  
[Оценка ответа](#)  
    [Логика проставления оценок](#)  
    [Оценить](#)  
    [Просмотреть оценки](#)  
[Уточнение к вопросу или ответу](#)  
[Закрытие платного вопроса](#)  
[Функционал жалоб](#)  
[Оценка в AppStore](#)  
[Меню](#)  
    [Как работает меню](#)  
[Вопросы на проекте](#)  
    [Страница вопроса](#)  
[Вход/Регистрация](#)  
    [Экран входа](#)  
    [Экран регистрации](#)  
    [Работа кнопок социальных сетей](#)  
    [Как работает клавиатура](#)  
    [Восстановление пароля](#)  
[Настройки](#)  
    [Настройки](#)  
    [Информация о пользователе \(для авторизованного\)](#)  
[Обратная связь](#)  
[Расскажите о нас](#)  
[Прочие страницы](#)  
    [Открытие ссылок. Внутренний браузер](#)  
    [Просмотр фото](#)  
[Push-уведомления](#)  
[Системные события](#)  
    [Валидация полей ввода](#)  
    [Лоадер](#)  
    [Подача вопроса из приложения и сбор лидов](#)  
    [Рассылка писем](#)  
    [Отсутствие интернета](#)  
[Дополнительные комментарии по исходным дизайн-макетам](#)

## Введение

Настоящий документ (далее по тексту - "Техническое задание") является приложением к Договору от 20 марта 2015 г. (далее по тексту - "Договор").

В "Техническом задании" приводятся требования к разработке "Приложения" и выполнения смежных работ, связанных с разработкой и выполнением обязательств по Договору. Подпись "Заказчика" и "Исполнителя" на настоящем документе подтверждает согласие сторон с нижеследующими фактами и условиями:

"Заказчик" подготовил и разработал "Техническое Задание", которое содержит перечень требований к выполняемым "Исполнителем" работам.

"Исполнитель" согласен со всеми положениями "Технического Задания".

"Заказчик" вправе требовать от "Исполнителя" выполнения работ и оказания услуг в полном соответствии с требованиями, описанными в "Техническом задании".

"Исполнитель" обязуется выполнить работы в объеме и в соответствии с требованиями, указанным в "Техническом Задании".

Все неоднозначности и неточности, выявленные в "Техническом задании" после его подписания, подлежат двухстороннему согласованию между "Сторонами".

В случае возникновения объективных причин, по которым не могут быть выполнены требования, описанные в "Техническом задании", изменение требований подлежит двухстороннему согласованию между "Сторонами".

В процессе согласования неоднозначностей, неточностей и изменений в "Техническом задании" могут быть разработаны дополнительные требования, которые оформляются дополнительным соглашением к "Договору" и соответствующим образом оцениваются.

## Макет

Предоставляется Заказчиком по запросу.

## Аппаратные и программные требования

Приложение должно запускаться и быть доступно для скачивания в AppStore для следующих устройств:

Все модели iPhone с версии 4 (включительно)

Все модели iPad и iPad Mini

Минимальная версия программного обеспечения iOS 7.

## Требования к коду приложения

Код приложения должен соответствовать Guideline от Apple: [Apple](https://developer.apple.com/library/ios/documentation/Cocoa/Conceptual/ProgrammingWithObjectiveC/Conventions/Conventions.html)  
(<https://developer.apple.com/library/ios/documentation/Cocoa/Conceptual/ProgrammingWithObjectiveC/Conventions/Conventions.html>).

Приложение должно быть реализовано на Objective-C.

Подключаемые сторонние программные решения (библиотеки, фреймворки и т.д.), которые не входят в пакет XCode, должны быть согласованы с заказчиком.

Код приложения (классы и методы) должен быть документирован в соответствии с требованиями guideline от Apple:  
<https://developer.apple.com/library/mac/documentation/DeveloperTools/Conceptual/HeaderDoc/intro/intro.html>

## Работа с API

Взаимодействие с Правовед.RU осуществляется через REST API по адресу <https://pravoved.ru/rest/>.

### Основные сведения

При составлении запросов необходимо оперировать сущностями (документами) и действиями над документами.

Сущность определяется в URL, действие - в поле Method HTTP заголовка запроса.

Поле Method может принимать следующие значения:

- GET - получение списка сущностей или конкретной сущности
- POST - создание новой сущности
- PUT - изменение существующей сущности
- DELETE - удаление сущности.

### Примеры запросов

GET <https://pravoved.ru/rest/users/self/> - *получение информации о текущем пользователе*

GET <https://pravoved.ru/rest/users/345/> - *получение информации о пользователе с ID 345*

POST <https://pravoved.ru/rest/questions/> - *создание нового вопроса*

GET <https://pravoved.ru/rest/questions/> - *получение ленты вопросов*

GET <https://pravoved.ru/rest/questions/567/> - *получение вопроса с ID 567*

PUT <https://pravoved.ru/rest/questions/567/> - *редактирование вопроса с ID 567*

DELETE <https://pravoved.ru/rest/questions/567/> - *удаление вопроса с ID 567*

Первым словом заглавными буквами описывается значение поля метода (GET, POST, PUT, DELETE). Далее написан полный или относительный адрес документа (<https://pravoved.ru/rest/users/self/> или [/rest/users/self/](#)). Если путь относительный, то подразумевается хост <https://pravoved.ru>.

Для всех запросов есть обязательный заголовок `Accept` с установленным значением `application/json`. Далее в ТЗ это поле не будет указано, но оно должно обязательно присутствовать.

Некоторые документы могут быть доступны только авторизованным пользователям. Для доступа к таким документам обязательно нужно добавить заголовок `Authorization` со значением `"Bearer AUTH_TOKEN"` (подробнее об авторизации написано ниже). Как и `Accept`, `Authorization` не указывается в документации. Запросы, которые требуют наличие этого заголовка помечены фразой `"Требуется авторизация"`.

### Пример запроса авторизации пользователя

Rest метод	Параметры	Результат
POST <a href="#">/rest/auth/</a>	<b>email</b> - string - email пользователя, ключ для которого необходимо получить. Пользователь должен быть зарегистрирован.  <b>password</b> - string - пароль пользователя	<pre>{   "meta":   {     "code": 200,     "message": "OK"   },   "data":   {     "status": "success",     "access_token": "RjQzRT...UU1",     "user_info":     {       "id": "792",       "usertype_id": "2"     }   } }</pre>

Метод: POST

Адрес документа: <https://pravoved.ru/rest/auth/>

Тело запроса: email=(string)&password=(string)

Для запросов с методами POST, PUT, DELETE параметры необходимо передавать в поле Body, для запросов с методом GET параметры являются частью адреса документа

## Пример запроса информации о текущем пользователе

Получение токена описано в предыдущем примере, токен будет возвращен в случае успешной авторизации.

Приватный запрос формируется по тем же правилам, что и публичный, кроме того, что добавляется заголовок "Authorization: Bearer AUTH\_TOKEN"

Rest метод	Параметры	Результат
GET /rest/users/self/	Требуется авторизация	<pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "users": [       {         "id": "792",         "email": "zubrillin@gambit.ru",         "name": "Повторка",         "surname": "Повторка",         "lastname": "",         "usertype_id": "2",         "role": "client",         "company_id": null,         "is_company_admin": null,         "euser_id": "792",         "city_id": "8",         "city_name": "Москва",         "city_label": "Г",         "is_online": null,         "is_online_till": "0",         "rating": 0,         "total_consultations": null,          "positive_client_feedbacks_percent": 0,         "balance": "null",         "avatar":         "/avatar_male_client_default.jpg"       }     ]   } }</pre>

Метод: GET

Адрес документа: <https://pravoved.ru/rest/auth/>

Значение заголовка Authorization: Bearer  
NjkwN0Y2NjQtN0JCMY0wRDI1LTlzRDYtOTBDRTQ3NzUxRDgw

## Ответ сервера

Ответ сервера является json объектом

```
{
  "meta": {
    "code": 200,
    "message": "OK"
  },
  "data": {
    "users": [
      {
        "id": "792",
        "email": "zubrilin@gambit.ru",
        ...
      }
    ]
  }
}
```

Обязательные поля ответа: meta и data.

В поле meta содержатся мета-данные об ответе. Обычно это код (code) и текстовое сообщение о результате выполнения. Code 200 означает успешное исполнение запроса. Числовые значения поля code соответствуют [состояниям HTTP запросов](#).

В поле data возвращаются данные результата исполнения запроса. Данные и формат поля data зависят от контекста запроса.

Пример ответа о неправильном POST запросе:

```
{
  "meta": {
    "code": 200,
    "message": "OK"
  },
  "data": {
    "status": "error",
    "errors": [
      {
        "message": "Неправильный логин или пароль",
        "type": "element",
        "name": "password"
      }
    ]
  }
}
```

```

    ]
  }
}

```

В поле **errors** содержится массив объектов ошибок. Поля ошибок:

**message** - текстовое сообщение об ошибке

**type** - тип сущности к которой относится ошибка: **element** или **form**. Указывает на то, ошибка касается какого-то одного элемента формы с запросом, либо всей формы.

**name** - необязательное поле - имя элемента формы, к которому относится ошибка.

## Batch-запросы

Чтобы уменьшить количество запросов к серверу, можно объединить несколько GET запросов в один.

Запрос представляет собой json от hash map:

```

array (
  array (
    'method' => 'GET',
    'relative_url' => '/rest/users/self/',
  ),
  array (
    'method' => 'GET',
    'relative_url' => '/rest/auth/',
  ),
  array (
    'method' => 'GET',
    'relative_url' => '/rest/static/lawyer-intro/',
  ),
)

```

Есть ограничение по количеству объединенных запросов: их должно быть не больше 5. Суммарное время исполнения запросов должно быть не больше секунды. Если запрос выполняется дольше, то результат будет содержать ответы на запросы, получение которых было меньше секунды в сумме, а на остальные запросы будет возвращен **null**.

*Таблица 1.1. Batch запросы*

Rest метод	Параметры	Результат
POST /rest/batch/	<b>batch</b> - string - json от массива с перечислением необходимых запросов.	Массив из ответов на каждый запрос. <pre> array (   0 =&gt;     array (       'meta' =&gt;         array ( </pre>



		<pre>         'code' =&gt; 200,         'message' =&gt; 'OK',     ),     'data' =&gt;     array (         'users' =&gt;         array (             0 =&gt;             array (                 'id' =&gt; '792',                 'email' =&gt;                 'zubrilin@gambit.ru',                 ...             ),         ),     ),     1 =&gt;     array (         'meta' =&gt;         array (             'code' =&gt; 0,             'message' =&gt; 'Not implemented',         ),         'data' =&gt; NULL,     ),     2 =&gt;     array (         'meta' =&gt;         array (             'code' =&gt; 200,             'message' =&gt; 'OK',         ),         'data' =&gt;         array (             'html' =&gt; '&lt;h1&gt;Привет, клиент!&lt;/h1&gt;',         ),     ), ) </pre>
--	--	--

## Авторизация пользователя

Авторизация требуется для того, чтобы приложение получило авторизационный токен и в дальнейшем Правовед мог идентифицировать по этому токenu автора запросов.

Таблица 1.2. Получение авторизационного токена

Rest метод	Параметры	Результат
POST /rest/auth/	<b>email</b> - string - email пользователя, ключ для которого необходимо получить. Пользователь должен быть зарегистрирован.  <b>password</b> - string - пароль пользователя	<pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "success",     "access_token": "RjQzRT...UU1",     "user_info": {       "id": "792",       "usertype_id": "2"     }   } }</pre>

### Разлогинивание пользователя

Для разлогинивания (например, если пользователь приложения хочет авторизоваться под другим аккаунтом) следует использовать метод DELETE /auth/. При этом удаляется авторизационный токен на сервере.

Таблица 1.3. Удаление авторизационного токена

Rest метод	Параметры	Результат
DELETE /rest/auth/	Требуется авторизация	<pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "success"   } }</pre>

### Регистрация устройства для push-уведомлений

Таблица 1.4. Регистрация устройства

Rest метод	Параметры	Результат
POST /rest/push-message-registration/	<b>registration_id</b> - string - id устройства для регистрации. <b>service_type</b> - string - код сервиса для рассылки push	<pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "success"   } }</pre>

	уведомлений. Для iOS этот параметр должен быть равен "ios".  Требуется авторизация	<pre>} }</pre>
--	--	--------------------

Таблица 1.5. Отмена регистрации устройства

Rest метод	Параметры	Результат
DELETE /rest/push-message-registration/	<b>registration_id</b> - string - id устройства для регистрации. <b>service_type</b> - string - код сервиса для рассылки push уведомлений. Для iOS этот параметр должен быть равен "ios".  Требуется авторизация	<pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "success"   } }</pre>

## Авторизация через сторонние OAuth сервисы

Общее описание взаимодействия мобильного приложения, серверного приложения и OAuth сервиса:

1. В мобильном приложении запрашивается доступ на получение токена доступа сервиса OAuth.
2. После получения токена происходит запрос на Правовед с передачей этого токена.
3. Правовед получает id пользователя OAuth сервиса и решает что с ним делать по схеме на рис. 1.
4. Если Правовед возвращает auth token, то клиентское приложение может совершать приватные запросы к API Правоведа.
5. Если правовед возвращает ошибку с сообщением о том, что не хватает данных о пользователе, это значит, что на серверном приложении нет пользователя с привязкой к OAuth аккаунту.
6. При получении ошибки следует сделать второй запрос с передачей данных о пользователе: email, имя, фамилия, тип пользователя (2 - клиент), пароль от аккаунта Правоведа, если email уже до этого регистрировался на Правоведе.

7. После получения данных происходит регистрация пользователя и привязка его к OAuth аккаунту. Если же пользователь был зарегистрирован, то осуществляется только привязка.

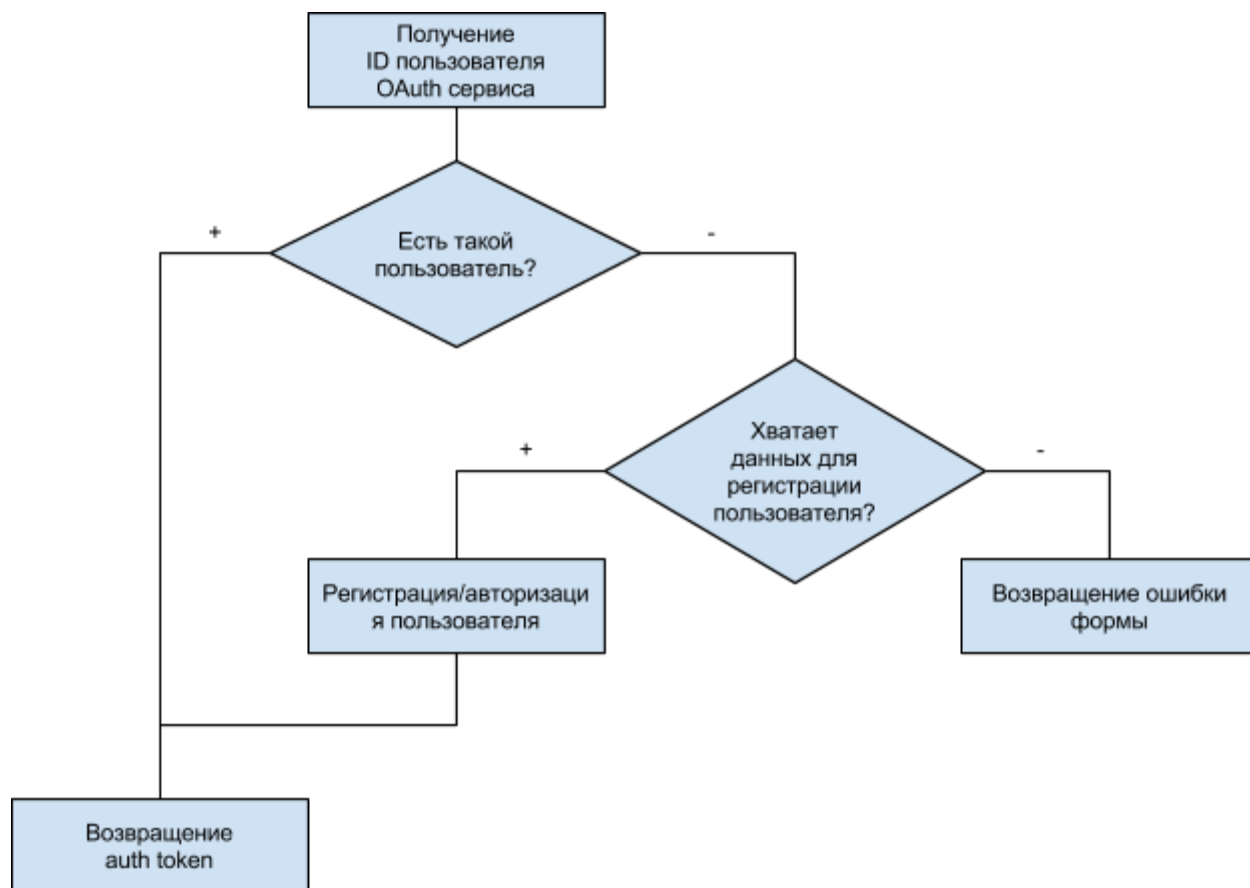


Рис. 1. Схема авторизации пользователя через OAuth

Таблица 1.6. Авторизация через OAuth

Rest метод	Параметры	Результат
POST /rest/oauth/	<b>service</b> - string - код сервиса авторизации. Возможные значения: vkontakte, facebook.  <b>access_token</b> - string - токен доступа OAuth сервиса.	Успешная авторизация пользователя <pre> {   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "success",     "access_token": "Mz.....zhC", </pre>

	<p><b>email</b> - string - необязательный параметр - email пользователя.</p> <p><b>password</b> - string - необязательный параметр - пароль, необходимо указать, если пользователь с таким email уже существует</p> <p><b>name</b> - string - необязательный параметр - имя пользователя.</p> <p><b>surname</b> - string - необязательный параметр - фамилия пользователя.</p> <p><b>usertype_id</b> - int - тип пользователя. Для клиента равен 2.</p>	<pre> "user_info": {   "id": "1320",   "usertype_id": "2" } } } </pre> <p>Нет привязки пользователя Правоведа к id пользователя OAuth сервиса</p> <pre> {   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "error",     "errors": [       {         "message": "Поле является обязательным для заполнения",         "type": "element",         "name": "email"       }     ],     "messages": "Unknown user",     "user_info": {       "name": "Constantine",       "surname": "Zubrilin",       "email": "",       "usertype_id": null     }   } } </pre>
--	---	---

## Список городов

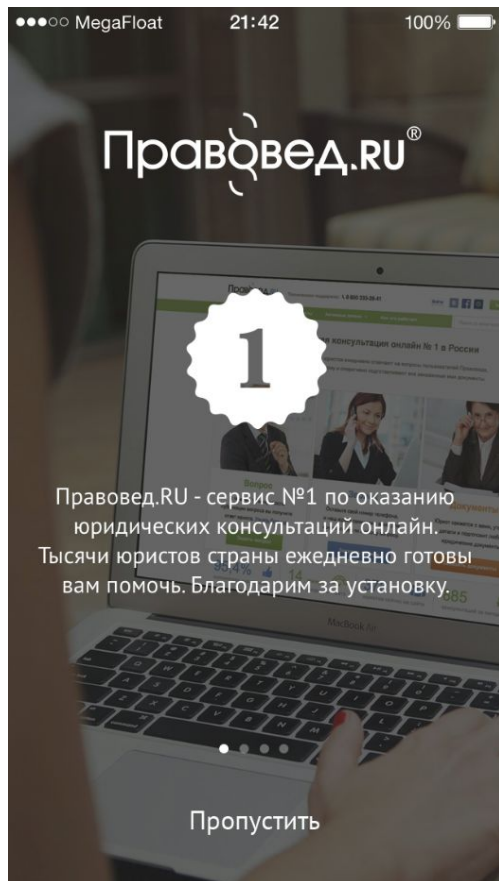
Для получения списка городов (в том числе и с фильтрацией по названию) необходимо сделать запрос, описанный в Таблице 2.1.

Таблица 2.1. Поиск города по названию

Rest метод	Параметры	Результат
GET /rest/cities/	<b>prefix</b> -string - первые буквы искомого города	<pre> {   "meta": {     "code": 200,     "message": "OK"   }, </pre>

	<p><b>additional_city</b> - int - id дополнительного города, который добавится в список в начале</p>	<pre> "data": {   "cities": [     {       "id": "16",       "name": "Санкт-Петербург",       "region_id": "78",       "area_id": null,       "label": "Город",       "short_label": "г",       "region": {         "id": "78",         "name": "Санкт-Петербург",         "label": "Город",         "short_label": "г"       }     },     {       "id": "198358",       "name": "Санки",       "region_id": "52",       "area_id": "1166",       "label": "Село",       "short_label": "с",       "region": {         "id": "52",         "name": "Нижегородская",         "label": "область",         "short_label": "обл"       },       "area": {         "id": "1166",         "name": "Лукояновский",         "label": "Район",         "short_label": "р-н"       }     }   ],   ... } </pre>
--	--	---

## Тutorial



При первой загрузке приложения пользователю показывается tutorial из 4 слайдов, которые можно листать. При повторных открытиях приложения tutorial не показывается (отображается экран загрузки).

Утвержденные слайды tutorialа находятся в исходных макетах.

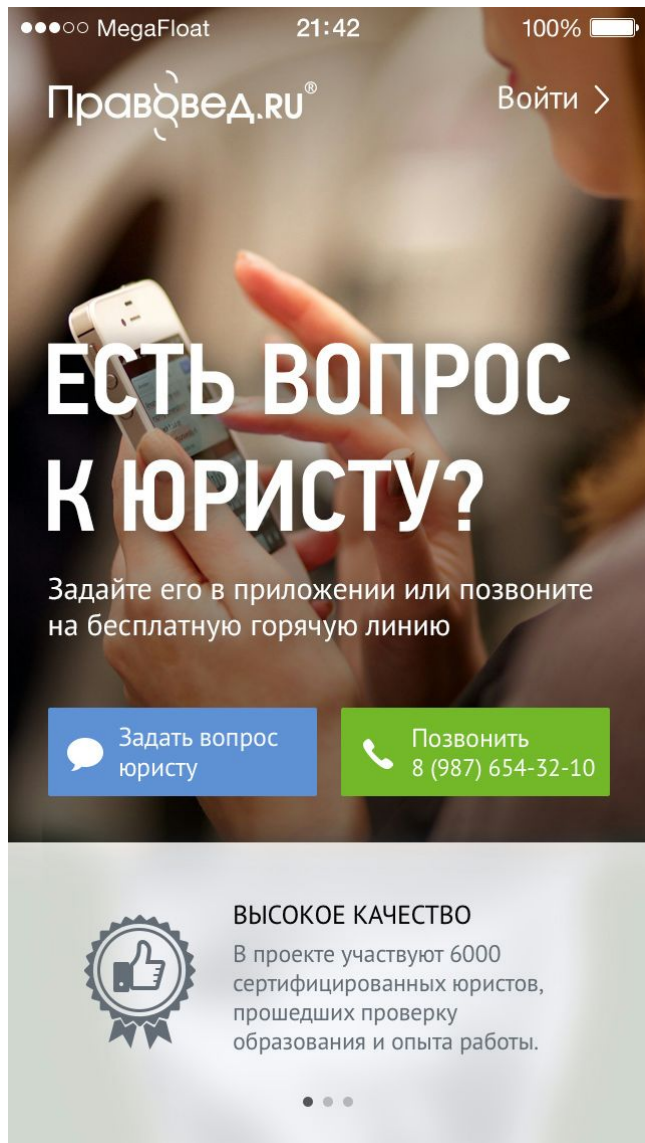
Слайд состоит из:

1. Логотип (только на первом слайде)
2. Фоновая фотография. У фотографии есть анимация движения. Изначальное положение, как в исходном макете. После открытия слайда фотография медленно движется влево. После достижения края, фотография начинает двигаться вправо. Диапазон движения 50 px в каждую сторону. Пример анимации можно посмотреть в приложении LinkedIn
3. Иконка
4. Текст

5. Кнопка “Пропустить”. При нажатии tutorial закрывается и открывается [главный экран](#).
6. На последнем слайде кнопка “Начать работу”. При нажатии tutorial закрывается и открывается [главный экран](#).

Состояние “tutorial просмотрен” сохраняется на клиенте.

## Главный экран



Перед каждой загрузкой приложения необходимо показывать launchscreen. В этот момент происходит загрузка данных приложения.

Таблица 3.2. Получение данных приложения с сервера



Rest метод	Параметры	Результат
GET /rest/static/		<p>Список статических страниц.</p> <pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "uri": [       "\rest\static\lawyer-intro\\"",       "\rest\static\promo-block\\"",       ...     ]   } }</pre>
GET /rest/options/		<p>Список настроек.</p> <pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "uri": [       "\rest\options\support-phone\\"",       "\rest\options\question-price\\"",       ...     ]   } }</pre>
GET /rest/options/question-price/		<p>Диапазон базовой стоимости платного вопроса и значение по-умолчанию:</p> <pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "paid_question_base_price_default": 600,     "paid_question_base_price_min": 100,     "paid_question_base_price_max": 10000   } }</pre>

GET /rest/options/payment-types/		<p>Способы оплаты вопроса:</p> <pre> {   "meta": {     "code": 200,     "message": "OK"   },   "data": [     "bill",     "BANKOCEAN2R"   ] } </pre>
-------------------------------------	--	---

Для получения списка популярных городов, необходимо сделать [запрос на получение городов](#) без указания префикса поиска и без дополнительного города.

Главный экран открывается при загрузке приложения для неавторизованного пользователя. Для авторизованного при загрузке приложения открывается страница ["Мои вопросы"](#).

Элементы:

1. Логотип
2. Кнопка-ссылка ["Войти"](#).
3. Текст. "Есть вопрос к юристу?" и "Задайте его в приложении или позвоните на бесплатную горячую линию"
4. Ссылка-кнопка "Задать вопрос юристу". При тапе открывается [форма подачи вопроса](#).
5. Для пользователей из Москвы и Санкт-Петербурга отображается кнопка с номером телефона (для каждого региона свой) с текстом: "Позвонить <номер>"
6. Внизу экрана отображается промо-слайдер. В блок информация должна попадать с сервера по запросу из Таблицы 2. Содержит изображение и текст. Слайдер имеет 3 (количество приходит с сервера) слайда, которые можно переключать движением слева направо. Также он меняет свое состояние самостоятельно (пролистывается влево через 5 секунд. Один слайд показывается 5 секунд).

Слайды:

- a. ВЫСОКОЕ КАЧЕСТВО. "В проекте участвуют 6000 юристов, прошедших проверку образования и опыта работы." 6000 - число юристов проекта
- b. ДОВОЛЬНЫЕ КЛИЕНТЫ. "95% клиентов, получивших юридическую консультацию на Правовед.RU, оставляют положительные отзывы."
- c. НАМ ДОВЕРЯЮТ. "Ежедневно сотни людей отправляют свои вопросы. Сегодня было дано 1400 консультаций". 1400 - число вопросов на проекте

### Таблица 3.3. Получение главного экрана

[illegible]

Перед загрузкой главного экрана пользователю выскакивает модальное окно с запросом местоположения.

- Текст: Для наиболее качественного оказания юридических услуг, разрешите определить ваше местоположение.
- Кнопка ОК. При нажатии модальное окно закрывается и на главном экране появляется телефон или не появляется в зависимости от региона.
- Кнопка Отмена. При нажатии модальное окно закрывается. На главном экране не появляется телефон.

## Форма подачи вопроса

## Точки входа

1. Главный экран
2. Пункт в меню
3. Кнопка на странице **“Мои вопросы”**, когда у пользователя еще нет вопросов

Вопрос пользователя всегда находится в форме (как черновик) до того момента, как он его задаст. Черновик сохраняется на клиенте.

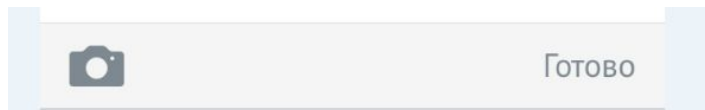
## Первый экран. Текст вопроса

Заголовок в шапке: “Шаг 1 из 3 Текст вопроса”

В шапке слева иконка Меню.

Элементы:

1. Поле ввода заголовков вопроса. Текст-подсказка: “Заголовок вопроса”. При нажатии в поле открывается экранная клавиатура. Русская раскладка. Кнопка-действия: “Далее”. При нажатии на “Далее” происходит перенос курсора в поле “Текст вопроса”. Наверху клавиатуры есть кнопка “Готово”, которая скрывает клавиатуру.
2. Поле ввода текста вопроса. Текст-подсказка: “Текст вопроса”. В поле Текст вопроса клавиатура имеет русскую раскладку. Кнопка-действия: “Готово”. По нажатию закрывает клавиатуру. Наверху клавиатуры есть кнопка “Готово”, которая скрывает клавиатуру и [иконка-фотоаппарат](#).



3. Кнопка “Далее.” Неактивна, пока пользователь не заполнил заголовок и текст вопроса. При показе клавиатуры скрывается под ней.

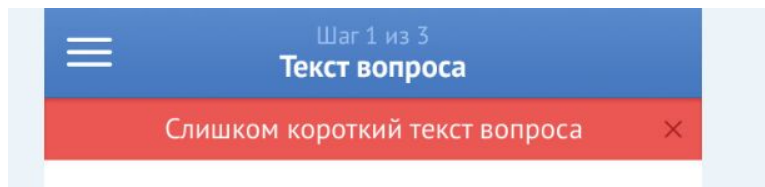
## Валидация

Текст вопроса

При нажатии на кнопку “Далее” происходит [валидация](#) длины текста вопроса. Длина **текста вопроса должна быть больше 100 символов.**

При ошибке:

“Слишком короткий текст вопроса.”



Заголовок

Также происходит валидация заголовка вопроса (**мин 5 символов и максимум 200; минимум два слова**).

Тексты валидации (приходят с сервера):

- “Слишком короткий заголовок. Введите минимум два слова” - если короткий заголовок
- “Слишком длинный заголовок вопроса” - если длинный заголовок

## Приложить фото

При клике на иконку-фотоаппарат открываются альбомы пользователя с возможностью выбора фото. В галерее в правом верхнем углу есть иконка-фотоаппарата, с помощью которой можно сделать фото и приложить его. Внизу кнопки “Готово” и “Отменить”

**При нажатии на Камера происходит запрос доступа к камере: “«Правовед.RU» запрашивает доступ к «Камере»”. После разрешения происходит съемка и если фото устраивает, то пользователь нажимает Использовать. При отказе в доступе к камере происходит возврат к модальной панели.**

**При выборе Галереи происходит такой же запрос как и к Камере. При разрешении доступа открывается Галерея пользователя с изображениями, которые можно выбрать и нажать Готово. Максимум установлен в 10 изображений.**

После добавления изображений их миниатюры помещаются после текста вопроса. При длительном нажатии на изображении, на них появляются крестики, при нажатии на которые изображения можно удалить.



Сохранение изображений на сервере осуществляется через API запрос, описанный в Таблице 3.1.

Для загрузки файла необходимо передавать его по частям. Параметр `file_id` идентифицирует части одного файла в рамках одной загрузки. Для получения `file_id` необходимо использовать запрос из таблицы 4.2. Рекомендуется разбивать файлы на части размером от 100 килобайт.

Сценарий загрузки:

1. Сабмит формы подачи вопроса
2. Если форма успешно подана, то клиент получает `user_id` и `question_id`
3. Клиент авторизуется и загружает файлы
4. После загрузки файлов делается запрос `PUT /questions/ID/` для привязки файлов к поданному вопросу (или черновика вопроса). Формат запроса в таблице 4.2а.

Таблица 4.1. Сохранение файла на сервере

Rest метод	Параметры	Результат
POST /rest/upload/	<p><b>file_upload_id</b> - string - Уникальный идентификатор для текущей сессии загрузки файла.</p> <p>Имя поля для загрузки: <b>file</b>.</p> <p>Заголовки: Content-type: multipart/form-data</p> <p>Имя файла: X-File-Name:example.jpeg</p> <p>Размер файла: X-File-Size:126615</p> <p>MIME-type файла: X-File-Type:image/jpeg</p> <p>Требует авторизации</p>	<p>Успешная загрузка части файла:</p> <pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "pending",     "file_upload_id": "rest_5503065be458f2.01477201",     "size": 204800   } }</pre> <p>size - количество байт файла, которые были успешно приняты и сохранены.</p> <p>Успешное завершение загрузки файла:</p> <pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "success",     "entity_id": 2631,     "file_upload_id": "rest_5503065be458f2.01477201",     "size": 648798   } }</pre>

Таблица 4.2. Получения id сессии загрузки файла

Rest метод	Параметры	Результат
GET /rest/upload/file_upload_id/	Требует авторизации	<pre>{   "meta": {     "code": 200,     "message": "OK"   }, }</pre>



Заголовок в шапке: “Шаг 2 из 3 О вас”

В шапке слева ссылка “Текст”.

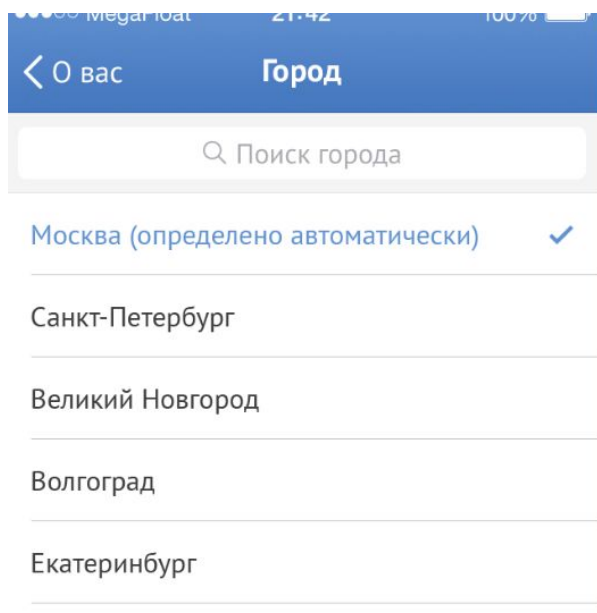
Элементы:

1. Поле ввода Ваше имя. Текст-подсказка: “Ваше имя”. При нажатии в поле открывается экранная клавиатура. Русская раскладка. Кнопка-действия: Далее. При нажатии на Далее происходит перенос курсора в поле Email. Наверху клавиатуры есть кнопка “Готово”, которая скрывает клавиатуру.
2. Поле ввода Email. Текст-подсказка: “Email”. При нажатии в поле открывается экранная клавиатура. Английская раскладка со значком @. Кнопка-действия: “Next”. При нажатии на “Next” происходит перенос курсора в поле Город. Наверху клавиатуры есть кнопка “Готово”, которая скрывает клавиатуру.
3. Поле с городом. Если человек определил свое местоположение на первом экране, то подставлять город, который был определен по GPS. **Если местоположение не определено то писать “Ваш город”**. По клику на поле открывается [экран выбора города](#).
4. Поле ввода Номер телефона. Текст-подсказка: “+7 (999) 888-77-66 ”. При нажатии в поле открывается экранная клавиатура. Цифровая раскладка. Кнопка-действия: “Готово”. При нажатии на “Готово” происходит закрытие клавиатуры. Наверху клавиатуры есть кнопка “Готово”, которая скрывает клавиатуру. Под полем телефона есть поясняющий текст: “По телефону с вами может связаться юрист для уточнения подробностей. Для вас звонок бесплатный. Номер нигде не публикуется.”
5. Кнопка “Далее.” Пока не заполнены все обязательные поля (имя, email, город) кнопка неактивна.

### Экран определения города

При нажатии в поле с городом происходит открытие экрана выбора города.



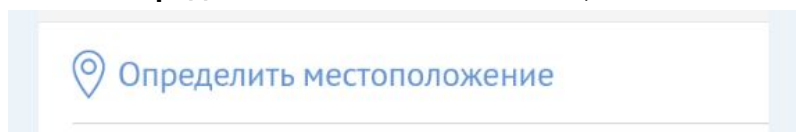


Заголовок в шапке: “Город”

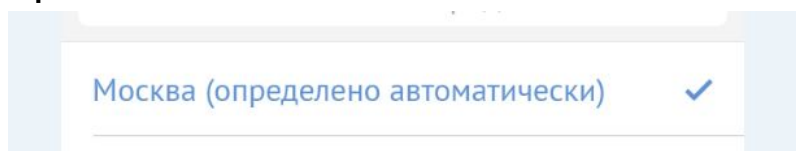
В шапке слева ссылка “О вас”.

Элементы:

1. Поле поиска. Текст-подсказка: “Поиск города” При тапе начинается поиск и открывается клавиатура. Русская раскладка. Кнопка действия: “Готово”. Как только пользователь начал вводить название города, происходит поиск по базе и подходящие города выводятся ниже. При вводе текста в правом углу строки поиска появляется кружок с крестиком, при нажатии на который строка поиска очищается и поиск сбрасывается.
2. Ссылка-кнопка “Определить местоположение”. При нажатии включается настройка “Определения местоположения”,



город пользователя автоматически определяется и подставляется вместо этой строки.



Если пользователь разрешил определять свое местоположение ранее, то вместо ссылки “Определить местоположение” сразу показывается город пользователя.

3. **Список предустановленных популярных городов (приходит с сервера при инициализации приложения).**
4. Чтобы выбрать другой город, пользователю необходимо нажать на искомое название, галочка переставится туда, экран выбора города закрывается, и выбранный город подставляется в соответствующее поле на странице “О вас”.

Для осуществления поиска по списку городов необходимо выполнить API [запрос на получение списка городов](#).

## Валидация

Происходит валидация после нажатия кнопки “Далее”:

1. Ввод некорректного адреса email. При ошибке появляется красный нотис: “Недопустимый адрес электронной почты.”
2. Ввод email юриста. При ошибке красный нотис: “Данный email принадлежит юристу.”
3. Ввод уже существующего email. Появляется дополнительно поле “Пароль” и сверху появляется зеленый нотис: “Данный email уже зарегистрирован. Введите пароль”. Если пароль правильный - переход на следующий экран. Если неправильный, то красный нотис: “Неправильный пароль”.

Валидация полей данного шага происходит на клиенте, кроме проверки email пользователя и пароля. Проверка пароля — API вызов аутентификации. В ответе приходит тип пользователя и дополнительно надо проверить, что email, введенный пользователем является email’ом клиента (usertype\_id = 2). После успеха пользователь становится авторизованным.

Для проверки email (связан-ли этот email с пользователем Pravoved.RU) необходимо использовать API метод, описанный в Таблице 4.3.

Таблица 4.3. Проверка email

Rest метод	Параметры	Результат
GET /rest/registration/is-vacant/	<b>email</b> - string - интересующий email	<pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "is_vacant": 1   } }</pre>

Ввод email и пароля должны сопровождаться запросом к API на авторизацию, описанный в разделе [Авторизация пользователя API](#).

### Третий экран. Тип вопроса

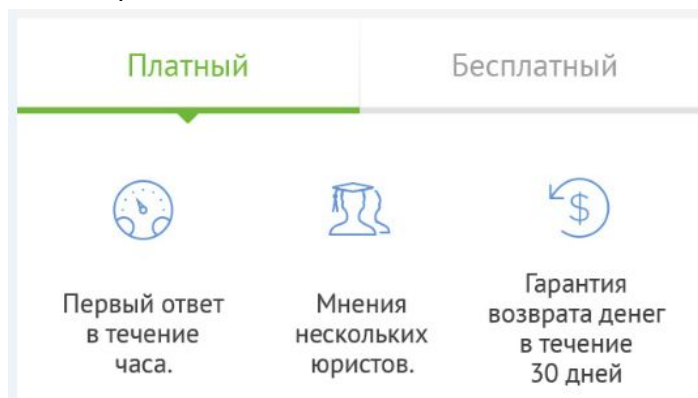
Заголовок в шапке: “Шаг 3 из 3 Тип вопроса”

В шапке слева ссылка “О вас”.

Segmented control с разделением на Платный и Бесплатный вопросы.

Платный (стоит по умолчанию)

1. Тростовые иконки с текстом



2. Стоимость вопроса

СТОИМОСТЬ ВОПРОСА

900 ₽



### ОБЫЧНЫЙ ВОПРОС

Ситуация, предполагающая уточнения и более детальное погружение в проблему.

- a. Указание выбранной стоимости. По умолчанию “600” (значение с сервера приходит при инициализации приложения)
  - b. Слайдер с диапазоном от 300 до 3000 рублей (значения с сервера приходит при инициализации приложения). Слайдер разбит на части: Легкий, Обычный, Сложный. При перемещении контрола отображается бабл с текущей стоимостью. По умолчанию стоит 600 рублей - обычный вопрос. Под слайдером отображается подсказка по текущей сложности вопроса:
    - i. Легкий вопрос. “Типовая ситуация, не требующая уточнений и подразумевающая краткий ответ.”
    - ii. Обычный вопрос. Ситуация, предполагающая уточнения и более детальное погружение в проблему.
    - iii. Сложный вопрос. Нетривиальная ситуация, допускающая противоречивость и требующая изучения документов.
3. Дополнительные опции. Список со свитчерами. Можно включить любое количество опций. Все опции по умолчанию выключены.

	<b>Приватный вопрос</b> +200 ₽ Не публикуется в открытом доступе, видят только юристы.	<input type="checkbox"/>
	<b>Срочный вопрос</b> +200 ₽ Гарантия ответа в течение 3 часов в будние дни 10.00-19.00 МСК.	<input checked="" type="checkbox"/>
	<b>Мнение эксперта</b> +200 ₽ Мнение эксперта проекта в дополнение к ответам юристов.	<input type="checkbox"/>

- a. Приватный вопрос. +200 рублей. “Не публикуется в открытом доступе, видят только юристы”
- b. Срочный вопрос. +200 рублей. “Гарантия ответа в течение 3 часов в будние дни 10.00-19.00 МСК.”

- с. Мнение эксперта. +200 рублей “Мнение эксперта проекта в дополнение к ответам юристов”
4. Количество отвечающих юристов. Текст подсказка: “Мнения нескольких юристов помогут вам избежать риска юридической ошибки”. Стандартный барабан с возможностью выбрать только одну из опций.

КОЛИЧЕСТВО ОТВЕЧАЮЩИХ ЮРИСТОВ

Мнения нескольких юристов помогут вам избежать риска юридической ошибки.

Один

Два

Три и больше +200 ₽

Опции:

- Один . Бесплатно. Выбрано по умолчанию
  - Два. +100 рублей
  - Три. +200 рублей
5. Итоговая стоимость вопроса. Складывается из стоимости вопроса, количества юристов и дополнительных опций.
6. Кнопка “Оплатить вопрос”

### Бесплатный

- Текст: “При подаче бесплатного вопроса гарантии ответа нет. Советуем вам выбрать платный вопрос.”
- Кнопка “Задать бесплатный вопрос”. При тапе вопрос задается и пользователю открывается экран с его вопросом. Пользователь авторизуется.

## Публикация бесплатного вопроса

API запрос на публикацию бесплатного вопроса описан в таблице 3.4.

Таблица 4.4. Подача бесплатного вопроса

Rest метод	Параметры	Результат
POST /rest/questions/	<p><b>title</b> - string - обязательное поле - Заголовок вопроса.</p> <p><b>text</b> - string - обязательное поле - Текст вопроса. Принимается и чистый текст и html.</p>	<p>Успешная подача вопроса:</p> <pre>{   "meta": {     "code": 200,     "message": "ОК"   },   "data": {</pre>

	<p><b>category_id</b> - int - обязательное поле - Категория вопроса. Значение 3 - без категории.</p> <p><b>city_id</b> - int - обязательное поле - Город к которому относится вопрос. -1 - город не указан.</p> <p><b>is_paying</b> - int - Платный или бесплатный вопрос (для бесплатного - 0).</p> <p><b>person_type</b> - string - Тип лица, который подает вопрос: частное лицо или юридическое. Значения: individual или corporate.</p> <p><b>phone</b> - string - Телефон.</p> <p><b>files[]</b> - array из int - Массив id файлов, которые будут прилинкованы к этому вопросу.</p> <p>Требуется авторизации</p>	<pre>         "status": "success",         "entity_id": "2201"       }     }   }    Ошибка в заполнении параметров:   {     "meta": {       "code": 200,       "message": "OK"     },     "data": {       "status": "error",       "errors": [         {           "message": "Заголовок           вопроса не может состоять из           одного-двух слов. Пожалуйста,           напишите заголовок, который           отражает суть вашего вопроса.",           "type": "element",           "name": "title"         },         {           "message": "Укажите тип           вопроса: бесплатный или           платный",           "type": "element",           "name": "is_paying"         },         {           "message": "Поле           является обязательным для           заполнения",           "type": "element",           "name": "category_id"         },         {           "message": "Не указан           тип вопроса",           "type": "element",           "name": "person_type"         },         {           "message": "Поле           является обязательным для           заполнения",           "type": "element",           "name": "city_id"         }       ]     }   } </pre>
--	--	---

		}
--	--	---

## Оплата вопроса

Заголовок в шапке: “Оплата вопроса”

В шапке слева стрелка назад.

На странице открывается веб-вью со страницей платежного шлюза для ввода данных банковской карты.

Пользователь вводит данные на странице шлюза, успешно оплачивает и попадает сразу на [страницу успеха оплаты](#) в нашем приложении. Промежуточной страницы успеха быть не должно.

**При неуспешной оплате, пользователю возвращается ошибка внутри веб-вью.**

## Последовательность API вызовов для публикации вопроса

Для публикации платного вопроса, в отличие от бесплатного, требуется выполнить несколько последовательных API вызовов. А именно:

1. Создание платного неоплаченного вопроса — Таблица 3.4
2. Передача реквизитов вопроса в платежную систему

Платежная система самостоятельно свяжется с сервером Pravoved.RU для передачи значения успеха выполненной транзакции. В этот момент произойдет публикация платного вопроса на сайте. Для того, чтобы уведомить мобильный клиент о публикации вопроса, необходимо периодически опрашивать API Pravoved.RU о статусе. Для этого следует [получить данные о вопросе](#) и проверить поле is\_published, в опубликованном вопросе оно будет равно 1.

В случае, если оплата не прошла, то вопрос станет черновиком на сайте с характерными [черновику свойствами](#).

Таблица 4.5. - Успешное принятие платного вопроса и ссылка для перехода на страницу оплаты

Rest метод	Параметры	Результат
POST /rest/questions/	<p><b>title</b> - string - обязательное поле - Заголовок вопроса.</p> <p><b>text</b> - string - обязательное поле - Текст вопроса.</p> <p><b>category_id</b> - int - обязательное поле - Категория вопроса. Значение 3 - без категории.</p> <p><b>city_id</b> - int - обязательное поле - Город к которому относится вопрос. -1 - город не указан.</p> <p><b>is_paying</b> - int - Платный или бесплатный вопрос (для платного - 1).</p> <p><b>phone</b> - string - Телефон.</p> <p><b>person_type</b> - string - Тип лица, который подает вопрос: частное лицо или юридическое. Значения: individual или corporate.</p> <p><b>payment_type</b> - string - Способ оплаты. bill (внутренний счет Правоведа) или, например,</p>	<p>Успешная отправка формы подачи вопроса с оплатой через сторонний сервис:</p> <pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "redirect",     "url": "\pay\external\?payment=payment_res ult_ec69488381e62f4775fcfe4fe6b85aeb",     "entity_id": "2202"   } }</pre> <p>Успешная отправка формы подачи вопроса с оплатой через Правовед:</p> <pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "success",     "entity_id": "2203"   } }</pre> <p>Ошибка валидации формы:</p> <pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "error",     "errors": [       {         "message": "Не указан тип вопроса",         "type": "element",         "name": "person_type"       },       {</pre>



	<p><b>BANKOCEAN2R</b> (внешняя система оплаты).</p> <p><b>clear_payment</b> - int - базовая стоимость вопроса, без дополнительных опций.</p> <p><b>lawyers_count</b> - int - Опция “количество юристов”. Значения: 1-3.</p> <p><b>is_private</b> - bool - Опция “приватность вопроса”. Значения: 0, 1.</p> <p><b>is_urgent</b> - bool - Опция “срочность вопроса”. Значения: 0, 1.</p> <p><b>council_answers_only</b> - bool - Опция “только ответы членов экспертного совета”. Значения: 0, 1.</p> <p><b>is_email_answers</b> - bool - Опция “уведомлять об ответах на email”. Значения: 0, 1.</p> <p>Все параметры обязательные.</p> <p>Требуется авторизация.</p>	<pre> "message": "Поле является обязательным для заполнения", "type": "element", "name": "is_urgent" } ] } } </pre>
--	--	---

## Экран успеха

Заголовок в шапке: “Успешная оплата”

Элементы:

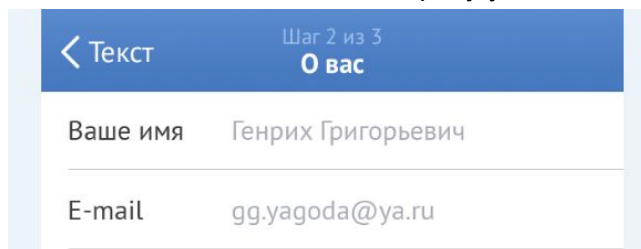
1. Заголовок “Ваш вопрос успешно оплачен”
2. Картинка
3. Текст: “В течение часа на него поступят ответы юристов.”
4. Кнопка “Перейти к вопросу”

## Для авторизованного пользователя

В форме подачи вопроса для авторизованного пользователя есть ряд изменений.

### Экран “О вас”

В полях ввода имени и email сразу указано Имя и email авторизованного пользователя.



Их нельзя изменить.

## Черновики

При создании вопроса в рамках приложения у пользователя всегда есть только один черновик, который он начал создавать в приложении. Черновик сохраняется в форме подачи вопроса и присутствует на [списке вопросов](#) с пометкой “черновик”.

При клике на такой вопрос, открывается форма подачи вопроса на том моменте, когда пользователь ушел с нее. Если пользователь ушел со страницы оплаты (не нашей), то мы возвращаем его на финальный шаг перед оплатой, чтобы он снова самостоятельно нажал “Оплатить вопрос”.

Черновик, созданный в приложении, мы передаем на сайт только в случае ошибки на странице оплаты (когда пользователь уже ушел на сторонний сайт для оплаты и не завершил ее).

Черновики, созданные на сайте, передаются в приложение при загрузке списков вопросов через API. Пользователь может завершить их в приложении. Промежуточная модификация черновиков на сервер не сохраняется.

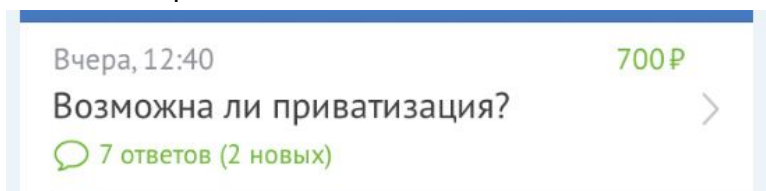
## Мои вопросы

Заголовок в шапке: Мои вопросы

В шапке слева иконка Меню.

На странице находится список вопросов, которые были заданы данным пользователем через сайт либо через приложение. Получить вопросы можно через API вызов, описанный в Таблице 8.

Сниппет вопроса состоит из:



1. Дата подачи вопроса. Дату необходимо указывать следующим образом:
  - a. Если задан сегодня, то указывать только время
  - b. Если задан вчера, то указывать “вчера 14:30”
  - c. Если задан в другой день, то указывать только число и месяц: “14 ноября”
2. Название вопроса (кликабельна вся ячейка). Если название слишком длинное (больше 3-х строк), обрезать его и ставить троеточие
3. Стоимость вопроса. Для платных вопросов
4. Иконка ответов и количество ответов на вопросе. Если на вопрос поступил ответ, а пользователь его еще не просмотрел, то иконка и число светятся зеленым. В скобках после всех ответов указывается количество новых. Как только пользователь открыл вопрос - ответы считаются прочитанными. Также при возникновении нового ответа происходит следующая нотификация:
  - a. Число новых ответов указывается на иконке приложения в iOS
  - b. Число новых ответов указывается на иконке меню в приложении
  - c. Число новых ответов указывается [напротив пункта “Мои вопросы”](#) в левом меню

Вопросы обновляются либо при оттягивании списка вниз (длинное движение) при этом должен отображаться лоадер оттягивания. Либо при каждой новой загрузке приложения

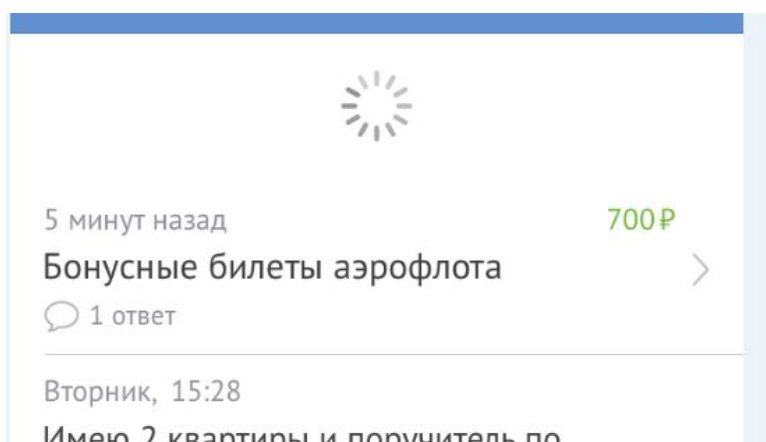


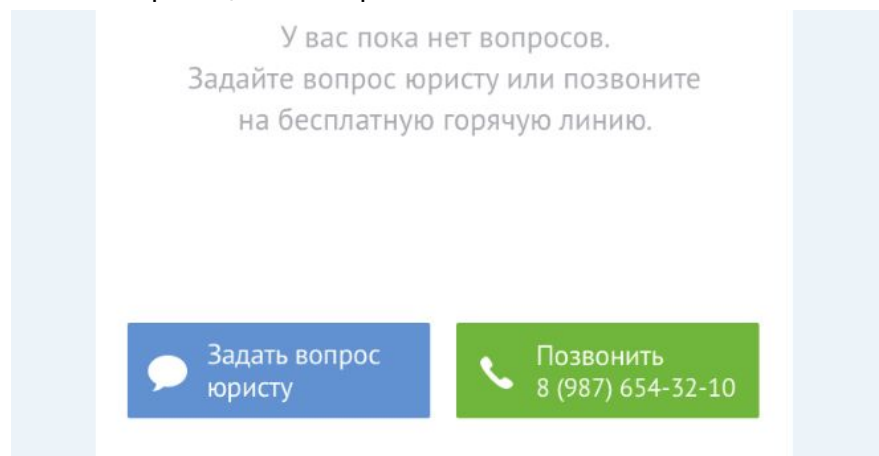
Таблица 5.1. Список вопросов пользователя

Rest метод	Параметры	Результат
GET /rest/questions/self/	<b>offset</b> - int - отступ от начала списка  <b>limit</b> - int - количество вопросов  Требуется авторизация	Массив вопросов пользователя <pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "questions": [       {         "id": "2204",         "title": "Взыскание алиментов со скрывающегося бывшего мужа",         "text": "&lt;p&gt;Как взыскать алименты со скрывающегося бывшего мужа?&lt;/p&gt;",         "short_text": "Как взыскать алименты со...",         "user": {           "id": "792",           "name": "Ольга",           "surname": "Петрова",           "usertype_id": "2",           "city_id": "8",           "city_name": "Москва",           "city_label": "Город"         },         "price": 900,         "price_for_lawyer": 630,         "time_published": "1425727491",</pre>

		<pre> "answer_count": "0", "is_for_contracted": "0", "required_answers": 0, "is_paying": "1", "is_closed": "0", "files": [  ], "question_uri": "\questions\2204\/", "answers_uri": "\questions\2204\answers\/", "question_comments_uri": "\questions\2204\question-comments\/" }, ... ] } } </pre>
--	--	--

Новые ответы должны приходить в приложение сразу же при их появлении в системе (аналогично сообщениям в Вконтакте). Пользователь не должен совершать никаких действий для получения новых ответов.

Если на странице нет вопросов:



1. Посередине текст (для СПб и Мск): “У вас пока нет вопросов. Задайте вопрос юристу или позвоните на бесплатную горячую линию”. Для других городов: “У вас пока нет вопросов. Задайте вопрос юристу”.
2. Кнопка “Задать вопрос юристу”. Открывает [форму подачи вопроса](#).
3. Если пользователь из Санкт-Петербурга или Москвы, то есть кнопка “Позвонить <номер телефона>”.

**Важно**

Получение информации о новых ответах, уточнениях или оценках происходит через систему Push нотификаций. Сам push содержит только ID вопроса, в котором произошли изменения.

## Страница вопроса


Заголовок в шапке: Вопрос №<номер вопроса в системе Правовед.RU>

В шапке слева стрелка назад.

Содержание страницы вопроса:

 **Вопрос № 768584**

**Регистрация проживания в доме на землях для дачного строительства**  
**Стоимость вопроса: 700 Р**  
Собственник построил таунхаус на три участка (3 свидетельства о праве собственности, 3 кадастровых паспорта), на землях населенных пунктов, разрешенное использование: для дачного строительства. Каждой... [развернуть](#)  
12 января 2014, 12:59  


 [Уточнить](#)  [Распределить гонорар](#)

1. Заголовок вопроса
2. Стоимость вопроса, если вопрос платный
3. Текст вопроса (5-6 строк с приложенными фото) с возможностью развернуть при клике на ссылку. В развернутом состоянии указываются вопрос полностью (весь остальной контент сдвигается вниз).
4. Дата подачи вопроса, [аналогично сниппету в списке](#).
5. Кнопка-ссылка “Уточнить”. По клику открывается [окно уточнения](#)
6. Ответы юристов: <кол-во ответов>. Если ответов юристов нет, писать текст “Ответы юристов поступят в ближайшее время. Вы получите уведомление о каждом ответе.”
7. **Ответ юриста:**



Татьяна Егорова

Юрист, Москва

Перевод нежилого помещения в жилое помещение не допускается, если такое помещение не отвечает требованиям.

12 января в 15:42

 Уточнить
 

 5
  0

- Аватар юриста
- Фамилия и Имя
- Город
- Ответ юриста (5 строк) с возможностью развернуть. В развернутом состоянии указываются ответ полностью (весь остальной контент сдвигается вниз). Если юрист приложил файлы, то они видны сразу без разворота. Система определяет если файл можно открыть, то на нем есть ссылка, если нет, то ссылки нет. Файл открывается в просмотрщике файлов, аналогично браузеру Safari. Важно исключить возможность воспроизвести XSS на стороне клиента при отображении содержимого приложенного файла через WebView.
- Время и дата ответа. Если ответ сегодняшний то писать только время, если ответ вчерашний, то перед временем писать “вчера”. У более старых ответов указывать сначала дату (без года) и время.
- Кнопка-ссылка “Уточнить”. По клику открывается окно уточнения. Для отправки уточнения на сервер необходимо использовать API метод, который описан в Таблице 10.
- [Кнопки оценки ответа](#) с количеством оценок

API получения информации о вопросе по ID

Таблица 5.1. Получение информации о вопросе и ответов на вопрос

Rest метод	Параметры	Результат
GET /rest/questions/:id/	<b>id</b> - int - id вопроса.  Требуется авторизация.	<pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "questions": [       {         "id": "2204",         "title": "Взыскание алиментов со скрывающегося бывшего мужа",</pre>

		<pre>         "text": "&lt;p&gt;Как взыскать алименты со скрывающегося бывшего мужа?&lt;/p&gt;",         "short_text": "Как взыскать алименты со...",         "user": {             "id": "792",             "name": "Ольга",             "surname": "Петрова",             "usertype_id": "2",             "city_id": "8",             "city_name": "Москва",             "city_label": "Г"         },         "price": 900,         "price_for_lawyer": 630,         "time_published": "1425727491",         "answer_count": "0",         "is_for_contracted": "0",         "required_answers": 0,         "is_paying": "1",         "is_closed": "0",         "can_i_answer": 0,         "can_i_accept_pro": 0,         "can_i_add_comment": 1,         "files": [          ],         "question_uri": "\questions\2204\/",         "answers_uri": "\questions\2204\answers\/",         "question_comments_uri": "\questions\2204\question-comments\/"     } } </pre>
GET /rest/questions/2204/answers/	Требуется авторизация.	<pre> {   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "answers": {       "1287": {         "id": "1287",         "question_id": "2167", </pre>



		<pre>         "question_uri":         "\question\2167\/",         "user_id": "1036",         "euser_id": "1036",         "text": "&lt;p&gt;В первую очередь надо собрать все документы&lt;/p&gt;",         "is_paid": 0,         "paid_percentage": null,         "feedbacks_count_positive": 0,         "feedbacks_count_negative": 0,         "user_can_rate": 1,         "timeadd": "1421931850",         "files": [          ],         "avatar":         "\images\design\avatar_male_lawyer _default.jpg",         "user": {             "id": "1036",             "name": "Платон",             "surname": "Кузнецов",             "city_id": "16",             "city_name": "Санкт-Петербург",             "city_label": "Г"         }     },     ... } } } </pre>
GET /rest/questions/:id/comme nts/	Требуется авторизация.	<pre> {   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "question_comments": {       "392": {         "id": "392",         "answer_id": null,         "text": "&lt;p&gt;123123 asd asdas dasdas&lt;/p&gt;",         "timeadd": "1425754337",         "files": [          ],         "question_uri": </pre>

		<pre> "/question/2204/"     }     }     }     } </pre>
--	--	--

API добавления уточнения к вопросу или ответу

*Таблица 5.2. Добавление уточнения к вопросу или ответу*

Rest метод	Параметры	Результат
<b>POST</b> /rest/question-comments/	<p><b>text</b> - string - обязательный параметр - текст уточнения.</p> <p><b>question_id</b> - int - обязательный параметр - id вопроса, на который добавляется уточнение.</p> <p><b>answer_id</b> - int - ответ, к которому добавляется уточнение.</p> <p><b>files[]</b> - array из int - Массив id файлов, которые будут прилинкованы к этому уточнению.</p> <p>Требуется авторизация.</p>	<pre> {   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "success",     "entity_id": "392"   } } </pre>

## Оценка ответа

При клике на плюс или минус открывается диалоговое окно с вариантами:

1. Оценить
2. Посмотреть оценки
3. Отменить

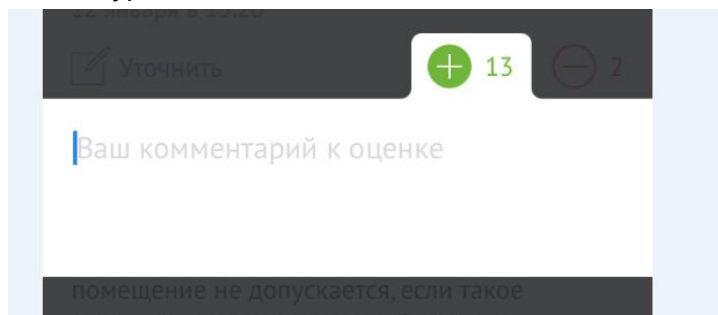
## Логика проставления оценок

Оценку ответу юриста в приложении может поставить только пользователь - автор вопроса, на который поступил данный ответ. Если он оценил ответ юриста (положительно или отрицательно), то этот ответ он больше оценивать не может.

Ответы юристов на чужих вопросах авторизованный пользователь оценить не может. Может только посмотреть оценки.

## Оценить

Прямо поверх других ответов появляется поле ввода комментария к оценке (экран затемняется, в правом верхнем углу появляется кнопка “Готово”) и открывается клавиатура.

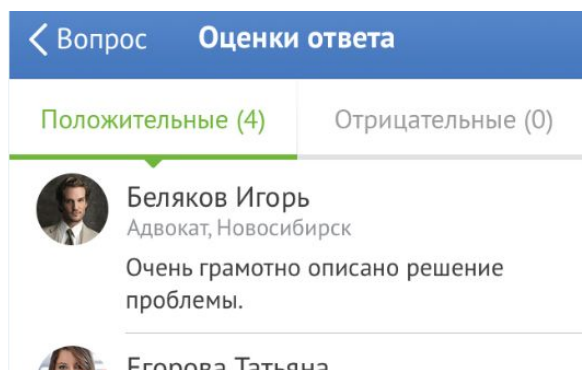


**Текст-подсказка в строке: “Ваш комментарий к оценке...”**

**Клавиатура на странице с русской раскладкой. Кнопка внизу-справа: “Готово”. При клике на кнопку ответ оценивается и клавиатура закрывается. Пока комментарий не введен кнопка на клавиатуре неактивна**

После успешной оценки та оценка, которую поставил пользователь принимает вид нажатой (кружок заливается) и число данных оценок увеличивается. Другая оценка принимает не нажатый вид (без кружочка).

## Просмотреть оценки



Открывается экран с информацией о том, кто поставил оценку.

1. Заголовок: “Оценки ответа”
2. Слева ссылка “Вопрос”
3. Вкладки “Положительные” и “Отрицательные”, в которые распределяются оценки. В скобках указано количество тех или иных оценок.
4. Аватар юриста или клиента
5. Фамилия и имя юриста или клиента
6. Статус: юрист, клиент, адвокат и прочее
7. Комментарий к оценке

Для получение списка оценок надо сделать запрос к серверу. Можно совместить с запросом на данные о вопросе с помощью [batch-запроса](#). Для сохранения новой оценки необходимо использовать API метод, который описан в Таблице 4.3.

Таблица 5.3. Добавление отзыва на ответ

Rest метод	Параметры	Результат
POST /rest/rate/	<p><b>entity</b> - string - обязательный параметр - тип сущности, должен быть равен "answer"</p> <p><b>entity_id</b> - int - обязательный параметр - id вопроса, на который оставляем отзыв</p> <p><b>text</b> - string - текст отзыва, обязателен, если отзыв отрицательный</p> <p><b>is_positive</b> - bool - обязательный параметр - при значении 1 - отзыв положительный, при 0 - отрицательный.</p> <p>Требуется авторизация.</p>	<pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "success",     "entity_id": "341"   } }</pre>

## Уточнение к вопросу или ответу

При нажатии на кнопку “Уточнение” открывается новый экран.

Заголовок в шапке: “Уточнение”

В шапке слева ссылка “Отмена”.

В шапке справа кнопка “Готово”. Пока текст не введен, кнопка неактивна.

Элементы:

1. Поле ввода текста уточнения. Текст-подсказка: “Текст уточнения”.

При открытии экрана “Уточнение” экранная клавиатура сразу открыта, и курсор находится в поле уточнения. Русская раскладка. Кнопка действия: “Готово”. Сверху клавиатуры есть иконка камеры.

## После добавления уточнения

### Уточнение

Технический паспорт на жилое строение (дача) один на весь дом. Мы хотим купить одну из трех частей таунхауса.

14 января, 01:15

Уточнение добавляется после основного текста вопроса или ответа. Заголовок “Уточнение”. Далее идет текст уточнения и фото, который выглядит [аналогично тексту вопроса](#). Метод добавления уточнения через API описан в Таблице 10.

## Закрытие платного вопроса

После появления хотя бы одного ответа юриста на платный вопрос, у клиента под вопросом появляется кнопка “Распределить гонорар”.

По тапу открывается “Экран распределения гонорара” (если ответило несколько юристов и это не рго-вопрос).

## Экран распределения гонорара

Отмена	Гонорар	Готово
Если ваш вопрос решен, и консультация оказана полностью, выберите от одного до трех юристов, которые вам помогли и распределите гонорар между ними.		
	Фролов Андрей	33% <input checked="" type="checkbox"/>
	Егорова Татьяна	33% <input checked="" type="checkbox"/>
	Григорьев Максим	33% <input checked="" type="checkbox"/>
	Беляков Игорь	<input type="checkbox"/>

Заголовок в шапке: Гонорар

В шапке слева ссылка “Отмена”.

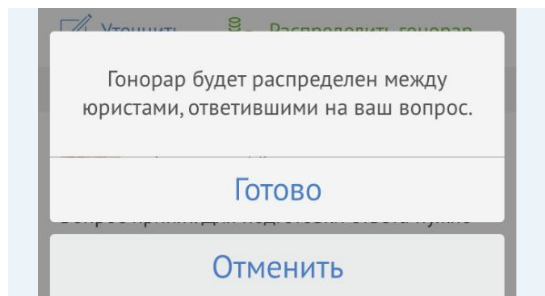
В шапке справа “Готово”:

1. При нажатии на нее происходит распределение гонорара.
2. Пользователя возвращает на страницу вопроса
3. Под вопросом текст “Вопрос решен”, вместо кнопки распределения гонорара
4. Рядом с ответом юриста текст с размером гонорара. Например: “Гонорар: 50%”

На экране:

1. Текст: “Если ваш вопрос решен и консультация оказана полностью, выберите от одного до трех юристов, которые вам помогли и распределите гонорар между ними.”
2. Сниметы юристов, ответивших на вопрос:
  - a. Аватар юриста
  - b. Фамилия
  - c. Имя
  - d. Свитчер, который включает распределение гонорара для данного юриста. После нажатия на свитчер, рядом с ФИО юриста высвечивается процент его гонорара. Гонорар всегда распределяется поровну между выбранными юристами. Максимум можно выбрать 3-х юристов. При выборе 3-х юристов другие юристы становятся неактивными.

Если на вопрос ответил один юрист или это Pro-вопрос



При нажатии на “Распределить гонорар” экран затемняется и снизу выезжает модальная панель с текстом: “Гонорар будет распределен между юристами, ответившими на ваш вопрос.”

Кнопки:

- Готово. Гонорар переходит одному юристу, либо распределяется между двумя, если это Pro-вопрос
- Отмена. Панель закрывается. Изменений не происходит.

Для распределения гонорара необходимо использовать API метод, который описан в Таблице 4.4.

Для закрытия платного вопроса надо указать: а) в массиве `paiduser_id` юристов б) в параметрах `paiduser_percent_ID` - процент, который получает конкретный юрист. Если ответ один, то выбранный юрист получит 100%. Если вопрос - ПРО, то не смотря на параметры, все юристы получают поровну.

Таблица 5.4. Закрытие вопроса и распределение гонорара

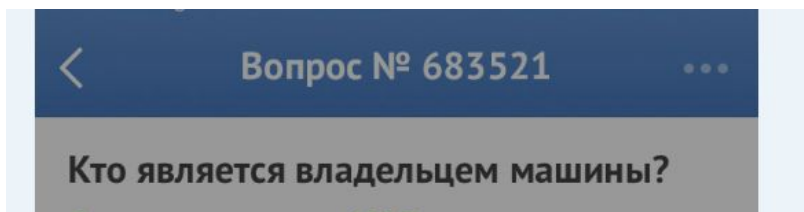
Rest метод	Параметры	Результат
PUT /rest/questions/:id/	<p><b>action</b> - string - обязательный параметр - для закрытия вопроса значение должно быть close.</p> <p><b>paiduser[]</b> - array из int - обязательный параметр - id пользователей, которым распределить гонорар. Для Pro-вопросов необязательный параметр.</p> <p><b>paiduser_percent_USERID</b> - int - обязательный параметр - процент гонорара, который перейдет юристу. USERID заменяется на id конкретного юриста. Значение параметра: от 1 до 100. Для Pro-вопросов необязательный параметр.</p> <p>Требуется авторизация.</p>	<pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "success"   } }</pre>

## Функционал жалоб

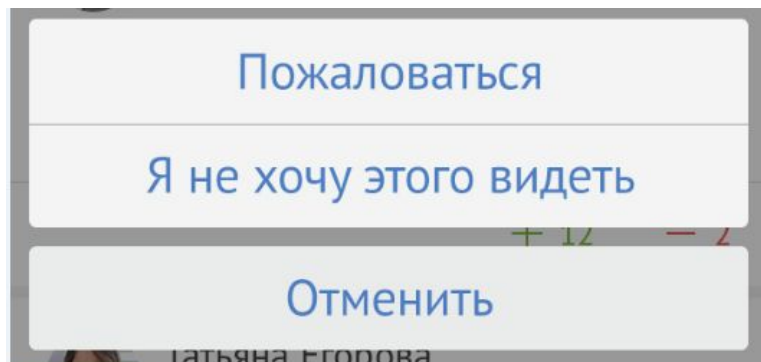
Данный функционал распространяется на чужие вопросы и ответы юристов (в своем и чужом вопросе), а также для авторизованного пользователя и нет.

### Жалоба на вопрос

На странице чужого вопроса (или любом для неавторизованного) в правом верхнем углу появляется троеточие,



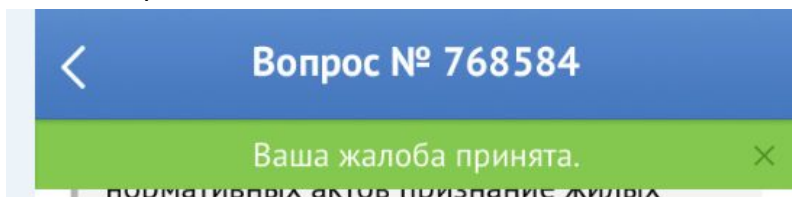
по нажатию на которое открывается модальное меню.



Состав меню (формируется на сервере):

- “Пожаловаться”. При нажатии открывается другое меню на месте этого со списком жалоб:
  - Спам
  - Оскорбление
  - Материал для взрослых
  - Пропаганда наркотиков
  - Детская порнография
  - Экстремизм
  - Насилие

При нажатии на тип жалобы меню закрывается, жалоба отправляется к нам на почту с помощью существующего API на отправку обратной связи с указанием (id сущности, типа жалобы). Наверху экрана появляется зеленый нотис об успешной отправке жалобы:



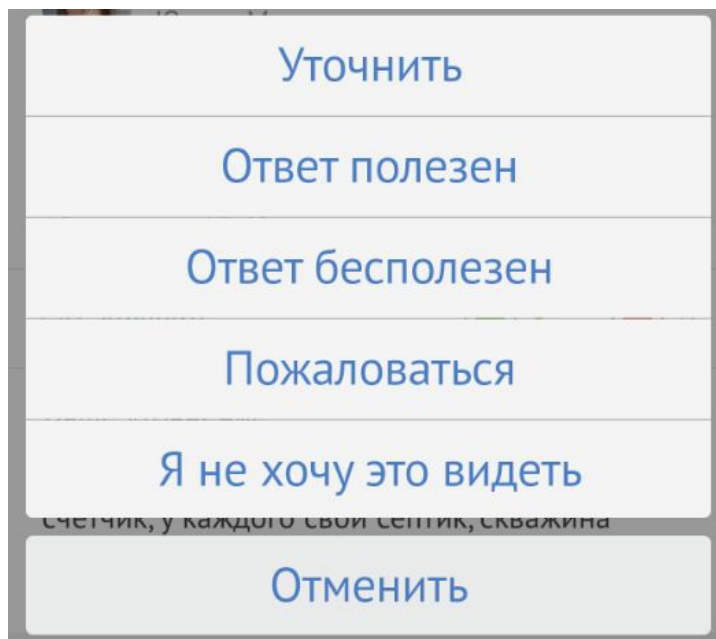
- “Я не хочу это видеть”. При нажатии на эту кнопку, происходит аналогичное действие как для кнопки Пожаловаться, только к сообщению добавляется пометка, что пользователь не хочет этого видеть.



Жалоба на ответ

Чтобы пожаловаться на ответ юриста необходимо нажать на него (на тело ответа).

После этого появляется модальное меню:



Частично меню дублирует функционал кнопок (ориентироваться на этот список ниже).

Меню для автора вопроса:

- [Уточнить](#)
- [Ответ полезен](#)
- [Ответ бесполезен](#)
- [Посмотреть оценки](#)
- “Пожаловаться”. Работает при нажатии как [на вопросе](#).
- “Я не хочу это видеть”. Работает также, как на вопросе.

Меню для пользователя на чужом вопросе:

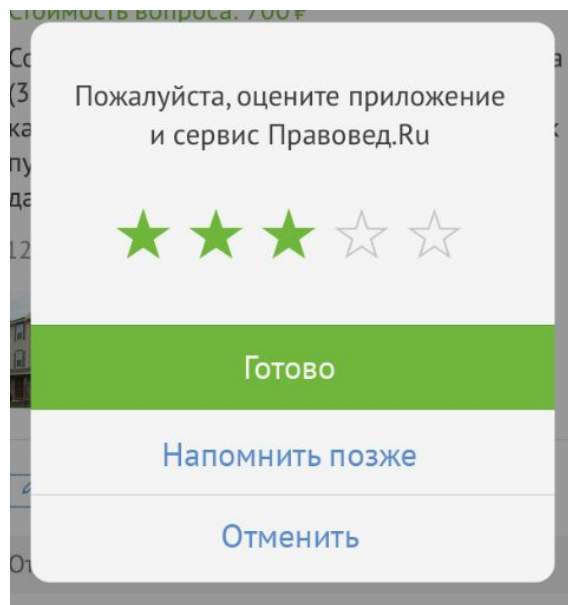
1. Посмотреть оценки
2. Пожаловаться
3. Я не хочу это видеть

Таблица 5.5. Жалоба на сущность

Rest метод	Параметры	Результат
POST /rest/feedback/	<b>text</b> - string - Текст жалобы. Текст генерируется на стороне клиента по одинаковому шаблону и	<pre>{   "meta": {     "code": 200,     "message": "OK"   }, </pre>

	<p>представляет собой объект в формате json.</p> <p>Объект жалобы содержит обязательные поля:  entity - сущность на которую жалуется пользователь;  question, answer, question_comment, rate;  entity_id - id сущности на которую жалуется пользователь;  reason - выбранная им причина жалобы.</p>	<pre>"data": {   "status": "success" }</pre>
--	---	--

## Оценка в AppStore



Сразу после распределения гонорара пользователю показывается модальное окно с просьбой оценить приложение в AppStore:

1. Текст: “Пожалуйста, оцените приложение и сервис Правовед.RU”
2. 5 звезд. По умолчанию установлены все звезды
3. Большая кнопка “Оценить”. Если поставлено 3 звезды и менее, то при нажатии, просто писать в том же окне: “Благодарим за оценку нашего приложения”. Если звезд 4 и более, то при клике открывается AppStore со страницей нашего приложения. Все поставленные в приложении оценки

необходимо передавать на сервер Pravoved.RU через API запрос, который описан в Таблице 5.6.

4. Ссылка “Напомнить позже”. При нажатии окно закрывается и появляется после следующего распределения гонорара
5. Ссылка “Отменить”. При нажатии окно закрывается и больше не показывается

Таблица 5.6. Сохранение оценки приложения

Rest метод	Параметры	Результат
POST /rest/app-rate/	<b>rate</b> - int - количество звезд. Значения: от 1 до 5.  Требуется авторизация.	<pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "success"   } }</pre>

## Меню

Меню доступно как авторизованному, так и неавторизованному пользователю.

Меню для неавторизованного пользователя:

1. Войти в профиль
2. Задать вопрос
3. Мои вопросы
4. Вопросы на проекте
5. Настройки
6. Обратная связь
7. Расскажите о нас

Для авторизованного:

1. Имя пользователя
2. Задать вопрос
3. Мои вопросы
4. Вопросы на проекте
5. Настройки
6. Обратная связь

В обеих версиях меню внизу есть кнопка звонка (для пользователей из СПб или Мск). На кнопке номер телефона (приходит с сервера) и текст “Бесплатная консультация”

## Как работает меню

Меню можно вызвать:

1. Свайпом слева (на всех экранах)
2. По тапу на иконку-гамбургер

Когда пользователь открывает меню в каком-либо разделе, данный раздел выделен в меню другим цветом.

## Вопросы на проекте

В данном разделе отображаются все платные вопросы, заданные на Правовед.

Заголовок в шапке: “Вопросы на проекте”

В шапке слева иконка меню.

В шапке справа иконка поиска

На странице отображаются последние платные вопросы со ссылками. Отображение [аналогично экрану “Мои вопросы”](#) Если вопрос приватный, то напротив него отображается иконка замка. При клике возникает модальное окно: “Вы не можете ознакомиться с данным вопросом, так как доступ к нему ограничен пользователем.”

**По умолчанию отображаются 20 вопросов (только платные).** При пролистывании экрана вопросы подгружаются с сервера. Во время подгрузки внизу отображается лоадер.

При нажатии на иконку поиска сверху экрана всплывает поле поиска с кнопкой “Найти”.

Также поиск можно вызвать по движению пальцем вверх. Одновременно с этим всплывает русская клавиатура, также содержащая кнопку “Найти”.

Пользователь вводит запрос и нажимает Найти. Происходит поиск (отображается лоадер). После чего выводятся результаты поиска. Клавиатура закрывается. Поиск происходит по всем вопросам с ответами. Если поиск не дал результата отображать: “Ничего не найдено”

Чтобы закрыть поиск, необходимо снова нажать на иконку “Поиска”.

Вопросы обновляются либо при оттягивании списка вниз (длинное движение) при этом должен отображаться лоадер оттягивания. Либо при каждой новой загрузке приложения.

Вывод списка вопросов:

*Таблица 6.1. Получения списка вопросов*

Rest метод	Параметры	Результат
------------	-----------	-----------

<p>GET /rest/questions/</p>	<p><b>offset</b> - int - отступ от начала списка</p> <p><b>limit</b> - int - количество вопросов</p> <p>Требуется авторизация.</p>	<pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "questions": [       {         "id": "2204",         "title": "Взыскание алиментов со скрывающегося бывшего мужа",         "text": "&lt;p&gt;Как взыскать алименты со скрывающегося бывшего мужа?&lt;/p&gt;",         "short_text": "Как взыскать алименты со...",         "user": {           "id": "792",           "name": "Ольга",           "surname": "Петрова",           "usertype_id": "2",           "city_id": "8",           "city_name": "Москва",           "city_label": "Город"         },         "price": 900,         "price_for_lawyer": 630,         "time_published": "1425727491",         "answer_count": "0",         "is_for_contracted": "0",         "required_answers": 0,         "is_paying": "1",         "is_closed": "0",         "files": [          ],         "question_uri": "\questions\2204\/",         "answers_uri": "\questions\2204\answers\/",         "question_comments_uri": "\questions\2204\question-comments\/"       },       ...     ]   } }</pre>
---------------------------------	--	--

Поиск вопроса

Таблица 6.2. Получения результата поиска

Rest метод	Параметры	Результат
GET /rest/questions/	<p><b>offset</b> - int - отступ от начала списка</p> <p><b>limit</b> - int - количество вопросов</p> <p><b>search</b> - string - строка запроса.</p> <p>Требуется авторизация.</p>	Аналогично с таблицей 6.1.

## Страница вопроса

Страница вопроса аналогична странице вопроса пользователя со следующими изменениями:

1. Нет кнопок “Уточнение”
2. Нельзя поставить оценки. Оценки можно только посмотреть. Сразу открывается просмотр при нажатии
3. **Соответственно нельзя распределять гонорар. Такой кнопки нет. Отображается только гонорар юристов и то, что вопрос решен.**
4. Под стоимостью вопроса указывается автор вопроса и его город.

## Вход/Регистрация

Изначально всегда открывается экран входа. Подробнее о технической реализации авторизации можно прочитать в разделе [Авторизация пользователя через API](#)

### Экран входа

Заголовок в шапке: Вход

В шапке слева иконка стрелки назад

Элементы:

1. Логотип Правовед.RU
2. Поле для ввода email. На поле серым текстом подсказка: “Email”
3. Поле для ввода пароля. На поле серым текстом подсказка: “Пароль”.
4. Кнопка “Войти”. Пока логин и пароль не введены, кнопка неактивна. При клике происходит авторизация пользователя в приложении. Открывается экран “Мои вопросы”.

5. Если введены ошибочные логин и пароль сверху появляется валидация: “Такой пары email/пароль не существует.” На поле валидации есть крестик для закрытия + данное поле само пропадает через 5 секунд.
6. Если введены пара логин и пароль для юриста, то нотис: “Пара email/пароль принадлежит юристу. Введите данные для аккаунта клиента”
7. Кнопки входа социальных сетей с текстом “Войти с помощью”:
  - a. Вконтакте
  - b. Фейсбук
  - c. **Одноклассники**
8. Ссылка “Забыли пароль?”
9. Ссылка-кнопка “Регистрация”. При клике открывается экран “Регистрация”.

## Экран регистрации

Заголовок в шапке: Регистрация

В шапке слева: Вход.

Элементы:

1. Поле для ввода имени. На поле серым текстом подсказка: “Ваше имя”
2. Поле для ввода email. На поле серым текстом подсказка: “Email”
3. Кнопка “Зарегистрироваться”. Кнопка неактивна пока не заполнены оба поля. При клике происходит авторизация пользователя в приложении. Открывается экран “Мои вопросы”. Пользователю на почту высылается письмо об успешной регистрации (24 шаблон) с его логином и паролем. При нажатии кнопки происходит валидация на правильность email. При ошибке появляется красный нотис валидации: “ Недопустимый адрес электронной почты.”
4. Текст: “Нажимая кнопку «Зарегистрироваться», вы принимаете условия [Пользовательского соглашения](#) и [Политики конфиденциальности](#).” При клике на ссылке открывается экран с крестиком в шапке, на котором находится просто отформатированный текст:
  - a. [Пользовательское соглашение](#).
  - b. [Политика конфиденциальности](#)

Если пользователь ввел email, который уже есть в системе, ему выводится сообщение об ошибке: “Такой email уже есть в системе. Воспользуйтесь формой входа или введите другой e-mail.”

API регистрации пользователя

Таблица 7.1. Регистрация пользователя

Rest метод	Параметры	Результат
------------	-----------	-----------

POST /rest/registration/	<p><b>email</b> - string - необязательный параметр - email пользователя.</p> <p><b>name</b> - string - необязательный параметр - имя пользователя.</p> <p><b>surname</b> - string - необязательный параметр - фамилия пользователя.</p> <p><b>usertype_id</b> - int - тип пользователя. Для клиента равен 2.</p>	<pre>{   "meta": {     "code": 200,     "message": "OK"   },   "data": {     "status": "success",     "access_token": "NEMzNUQ1RDEtQTZDMylCQjYxL Tc2NTktNEEyMjkyMDFFNzgw",     "user_info": {       "id": "1321",       "usertype_id": "2"     }   } }</pre>
-----------------------------	--	--

## Работа кнопок социальных сетей

При клике на кнопку социальной сети должно открываться приложение данной социальной сети на телефоне пользователя. В приложении пользователь должен дать согласие на использование своих личных данных приложением Правовед.RU.

Если у пользователя нет приложения выбранной социальной сети, либо у данной социальной сети нет функции авторизации через приложение, то мы внутри нашего приложения открываем web-view с предложением пользователю авторизоваться в необходимой социальной сети. После авторизации, пользователь дает свое согласие, как и при обычной авторизации через социальные сети.

После авторизации в соц. сети и согласия полученные данные идут в форму подтверждения. Если сервис не передает email, то пользователь должен будет его ввести дополнительно на промежуточном экране:

1. Текст: “Для успешного входа введите ваш email”
2. Фамилия Имя пользователя, полученные от соц. сети
3. Поле для ввода email
4. Кнопка “Войти”

**Если email пользователя уже есть в системе, высказывает зеленый нотис с предложением ввести пароль от этого email. Текст: “Указанный email уже есть в базе. Введите пароль”**

**Если email нет в базе данных, и email не совпадает с полученным от сервиса, то создаем новый аккаунт и направляем письмо с предложением активировать акаунт**



на указанный email. Идентификатор сервиса привязываем к аккаунту. Пользователя авторизуем в приложении.

## Как работает клавиатура

Изначально курсор не установлен в полях ввода.

При тапе на верхнее поле ввода открывается клавиатура, при этом содержимое экрана скрывается. На клавиатуре сверху всегда есть кнопка Готово, которая скрывает клавиатуру.

При клике на любую область экрана (не поле ввода) клавиатура скрывается.

Отличие клавиатуры в зависимости от поля ввода:

1. Email (экран входа). Английская раскладка со значком @. Нижняя правая кнопка: Next. При тапе на Next курсор переходит в поле “Пароль”
2. Пароль. Английская раскладка. Нижняя правая кнопка: Go. При тапе на Go происходит авторизация
3. Ваше имя. Русская раскладка. Нижняя правая кнопка: Далее. При тапе на Далее курсор переходит в поле Email
4. Email (экран регистрации и восстановления пароля). Английская раскладка со значком @. Нижняя правая кнопка: Go. При тапе на Go происходит авторизация и регистрация.

## Восстановление пароля

1. При клике на “Забыли пароль?” если у пользователя не был введен email открывается экран восстановления пароля (сверху слева - возврат к Входу).
  - a. Текст: Для восстановления пароля введите ваш email
  - b. Поле для ввода email.
  - c. Кнопка “Восстановить пароль” (неактивна пока пользователь не заполнил поле email). При нажатии кнопки происходит валидация на правильность email. При ошибке возникает красный нотис: “Недопустимый адрес электронной почты.” При прохождении валидации, пользователю отправляется письмо о восстановлении пароля. Для этого требуется выполнить API запрос, который описан в Таблице 17.
  - d. Открывается экран входа. Сверху всплывает зеленое сообщение: “На вашу электронную почту отправлено письмо о восстановлении пароля.”

Таблица 7.2. Восстановление пароля

Rest метод	Параметры	Результат
POST /rest/auth/restore/	<b>email</b> - string - обязательный параметр. Email аккаунта,	{ "meta": { "code": 200, }

	пароль к которому надо восстановить	<pre> "message": "OK" }, "data": {   "status": "success" } } </pre>
--	--	---

Если у пользователя был введен email на экране входа, и он нажал “Забыли пароль?”, то новый экран не открывается, и пользователь сразу получает сообщение об отправке восстанавливающего письма.

При переходе по кнопке в письме мы определяем мобильный пользователь или нет. Если мобильный, то показываем ему адаптированную страницу, которая содержит ту же информацию, что и сейчас.

После ввода и повтора паролей, пользователь попадает на мобильную страницу “Пароль успешно изменен”

1. Текст: “Дополнительные изменения в настройках профиля вы можете сделать в Личном кабинете.”
2. Кнопка “Продолжить на сайте” - открывает обычный сайт Правовед
3. Кнопка “Открыть iOS приложение”. Открывает приложение на мобильном, если оно установлено, если нет, то открывается AppStore с нашим приложением. Если это iOS система

## Настройки

На экране отображается информация о пользователе, который сейчас авторизован, а также базовые настройки.

Заголовок в шапке: Настройки

В шапке слева иконка Меню.

## Настройки

1. Push-уведомления. Контроль: Свитчер. При включении данной настройки пользователь будет получать от нас уведомления на свое устройство, независимо от типа уведомления. При выключении настройки уведомления приходить не должны.
2. Определение местоположения. Контроль: Свитчер. При включении данной настройки мы определяем местоположение пользователя. Для нас это важно, чтобы отображать релевантный номер телефона на главном экране. Выключение данной настройки для нас интереса не представляет. Мы перестаем определять местоположение, но на главный экран это никак не влияет.

## Информация о пользователе (для авторизованного)

Информация о пользователе (отображается только если пользователь авторизован):

1. Имя
2. Email

Далее идет текст: “Отредактировать свой профиль вы можете на сайте [Правовед.RU](http://pravoved.ru)”

На Правовед.RU находится ссылка, при клике на которую открывается внутренний браузер, пользователь авторизуется на pravoved.ru и открывается ссылка <http://pravoved.ru/private/settings/personal/>

Далее идет кнопка-ссылка “Выйти из профиля”

## Обратная связь

На экране отображаются телефоны и email для связи с технической поддержкой.

Заголовок в шапке: Обратная связь

В шапке слева иконка Меню.

Текст на экране: По техническим вопросам вы можете обратиться в службу поддержки пользователей.

Далее идут контактные данные с иконками:

1. 8 800 333-28-41 (для регионов). Иконка телефона
2. +7 812 424-15-21 (Санкт-Петербург). Иконка телефона
3. +7 499 705-84-25 (Москва). Иконка телефона
4. [support@pravoved.ru](mailto:support@pravoved.ru). Иконка письма

Контактные телефоны и электронную почту необходимо получать с сервера при инициализации приложения. Формат телефонов может изменяться (например, был +7 123 456-78- 90, а стал 456 78 90, либо (8 123) 456-78-90.

Номера телефонов и электронная почта кликабельны:

1. При клике на номер телефона происходит его набор и звонок.
2. При клике на email происходит открытие почтового сервиса, используемого по умолчанию и подставления указанного email в адресную строку.

## Расскажите о нас

Заголовок в шапке: Расскажите о нас

В шапке слева иконка Меню.

Элементы:

1. Картинка
2. Текст: “Благодарим за использование нашего приложения. Расскажите своим друзьям о сервисе Правовед.RU”
3. Кнопки социальных сетей.
  - a. Вконтакте
  - b. Одноклассники
  - c. Фейсбук

При клике открывается [внутренний браузер](#) с предложением рассказать о нас. Работа кнопок аналогична работе кнопок в футере нашего сайта.

## Прочие страницы

### Открытие ссылок. Внутренний браузер

Все гиперссылки на внешние источники должны открываться во внутреннем браузере приложения, который работает так:

1. Открывается новый экран. В шапке заголовок с названием страницы.
2. В левой части шапки крестик, который закрывает браузер.
3. В правой части иконка функций браузера
  - a. Открыть в Safari
  - b. Скопировать ссылку
  - c. Отменить

### Просмотр фото

При нажатии на фотографию, открывает просмотрщик:

1. Черный фон
2. Вверху слева “Закрыть”. Закрывает просмотрщик
3. Сверху посередине порядковый номер фото и общее количество фото. Например 1 из 3

Если логически есть несколько фото в одной сущности (например 5 фото в вопросе), то они просматриваются сразу все - пролистыванием.

## Push-уведомления

Рассылать пользователю уведомление, когда поступает ответ на его вопрос.

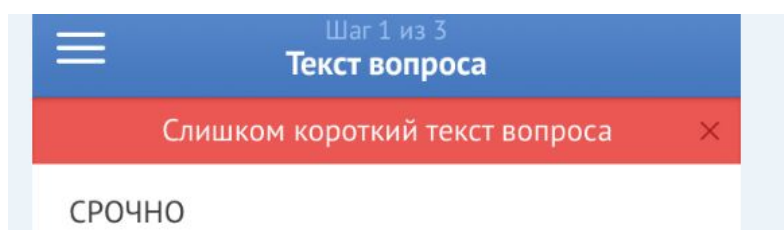
**В уведомлении:**

1. Иконка и название Правовед.RU
2. Текст: “Новый ответ юриста на ваш вопрос “Что мне делать?”

При нажатии на уведомление происходит открытие приложения на экране данного вопроса. Желательно скроллить экран до нового ответа юриста.

## Системные события

### Валидация полей ввода



При возникновении ошибки или неверного ввода значения, сверху появляется красный нотис с ошибкой.

В нотисе есть крестик, с помощью которого его можно закрыть. Также он сам закрывается через 5 секунд.

Нотис возникает только после нажатия на кнопку действия на странице.

### Лодер

При ожидании ответа от сервера экран блокируется и посередине возникает крутящийся лодер.

### Подача вопроса из приложения и сбор лидов

Необходимо на сервере реализовать возможность правильно идентифицировать:

1. Пользователей, зарегистрировавшихся в приложении
2. Вопросы, заданные пользователем через приложение

Также при подаче вопроса из приложения и оставления телефона мы собираем лиды и далее распространяем их по стандартной схеме.

### Рассылка писем

Рассылка писем при подаче вопроса и работе с ним через приложения аналогична рассылке при работе с вопросом на сайте.

Все письма должны рассылаться также.

## Отсутствие интернета

При отсутствии интернета один раз за сессию работы приложения выводить стандартное красное сообщение-ошибку сверху:

“Ошибка соединения

Проверьте свое подключение к интернету”

## Публикация “Приложения” в магазине приложений “App Store”

“Исполнитель” обязан обеспечить успешную публикацию “Приложения” в магазине приложений “App Store”. Принимая во внимание риски отказа в публикации по причине приема платежей внутри приложения, “Исполнитель” предпринимает все необходимые и достаточные меры, для успешной публикации “Приложения” в течении 2 месяцев с даты первой отправки “Приложения” в “App Store”.

## Дополнительные комментарии по исходным дизайн-макетам

Исходные и сопутствующие графические материалы предоставляются Заказчиком по запросу.

В архиве с исходными материалами организована следующая структура:

1. /640x960 — исходные материалы для модели iPhone 4 и 4S.
2. /640x1136 — исходные материалы для модели iPhone 5/5S/5C .
3. /750x1334 — исходные материалы для модели iPhone 6.
4. /1242x2208 — исходные материалы для модели iPhone 6 plus.
5. /fonts — файлы различных начертаний используемой гарнитуры PT Sans (распространяется свободно <http://www.paratype.com/public/>).
6. /photo — исходные изображения для слайдов Тutorials и Главного экрана.

Структура исходных материалов в папках для каждого разрешения:

1. login+settings — экраны авторизации, регистрации, настроек, просмотра файлов, обратной связи.
2. creation — экраны создания вопроса.

3. questions — экраны списков вопросов, экран вопроса в различных состояниях, распределение гонорара, оценка вопроса, оценка приложения.
4. tutorial — экраны Тutorials.
5. welcome+sidebar — Главный экран и Меню.

Дополнительная информация:

1. У всех элементов, которые подразумевают нажатие, в нажатом состоянии 50% прозрачности.
  2. В макете Главного экрана второй и третий слайды находятся в папках с отключенной видимостью (по структуре они находятся рядом с активным первым слайдом).
  3. Эффект "Блур" есть только на Главном экране под белой плашкой для слайдера.
- Данные: размытие по Гауссу, величина 20px.

От имени "Заказчика"

\_\_\_\_\_ Андреев С.В.

От имени "Исполнителя"

\_\_\_\_\_ Слижик А.В.