|  |  |
| --- | --- |
| IntraVision_small | ООО «Интравижн»  [cmail@intravision.ru](mailto:cmail@intravision.ru), [www.intravision.ru](http://www.intravision.ru)  8 (495) 795-23-44  Россия, Москва, ул. Б. Новодмитровская, д. 36 стр.1 |

# ТЕСТОВОЕ ЗАДАНИЕ НА ВАКАНСИЮ Front-End разработчика (React)

# Введение

Основная цель данного задания это понять «подход» кандидата к разработке и его желание и возможности самостоятельно разбираться в новых технологиях, добиваться поставленных целей.

Мы предполагаем, что кандидат имеет от одного года опыта FrontEnd разработки и имеет хорошие базовые знания web технологий.

# Задание на разработку SPA (single page application) Web приложения

Требуется разработать web-страницу по заданному дизайну (PSD – шаблон). Цель данной страницы просматривать список заявок, добавлять заявки и редактировать заявки. Мы разработали серверную часть (web API, спецификация ниже), а вам необходимо разработать Front часть.

## Описание поведения интерфейса WEB приложения

* Страница со списком заявок,

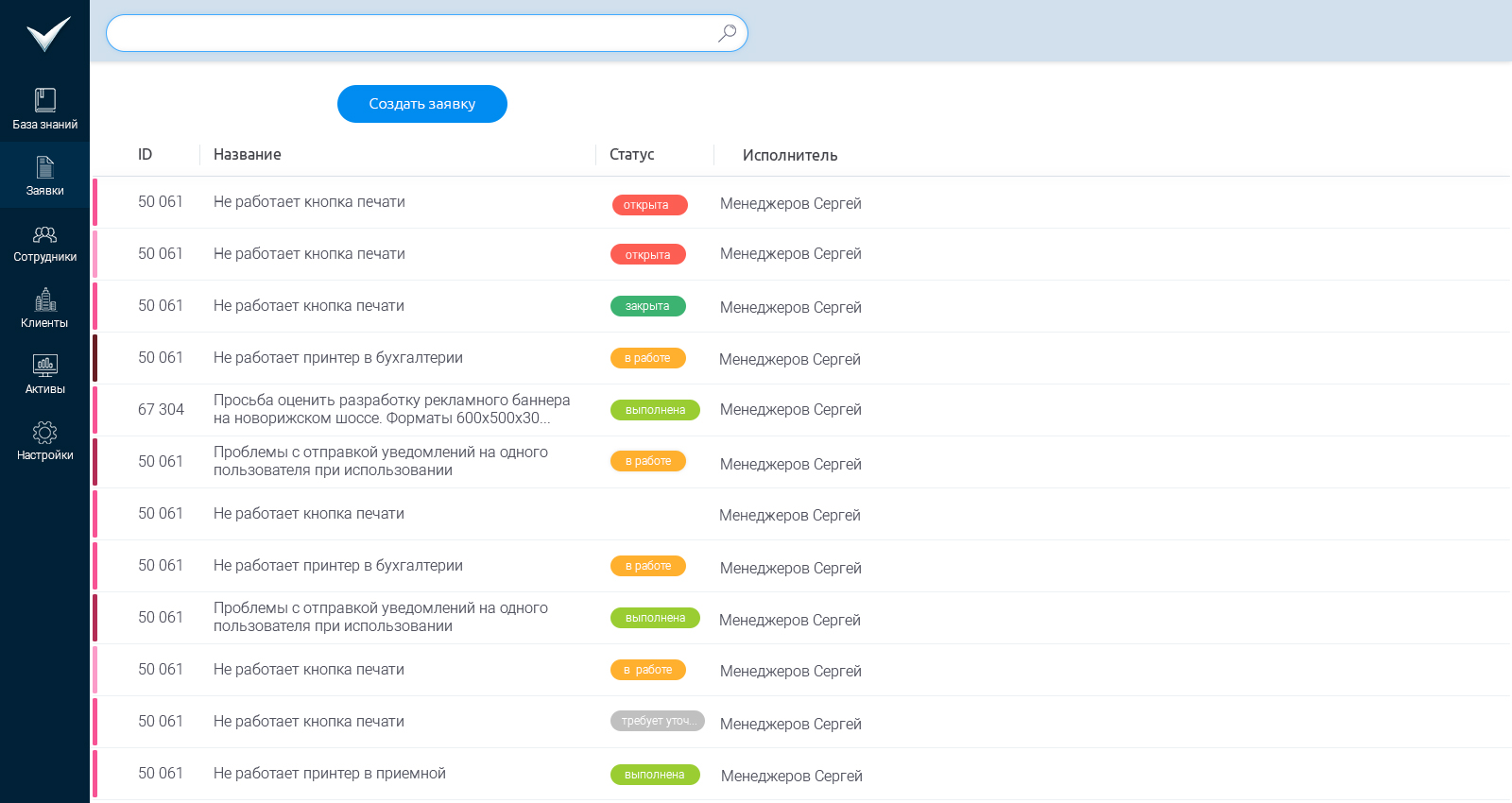
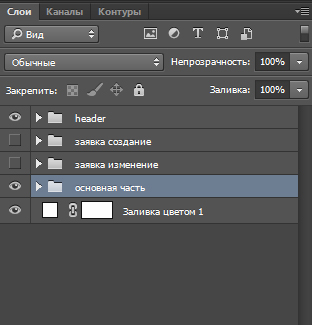


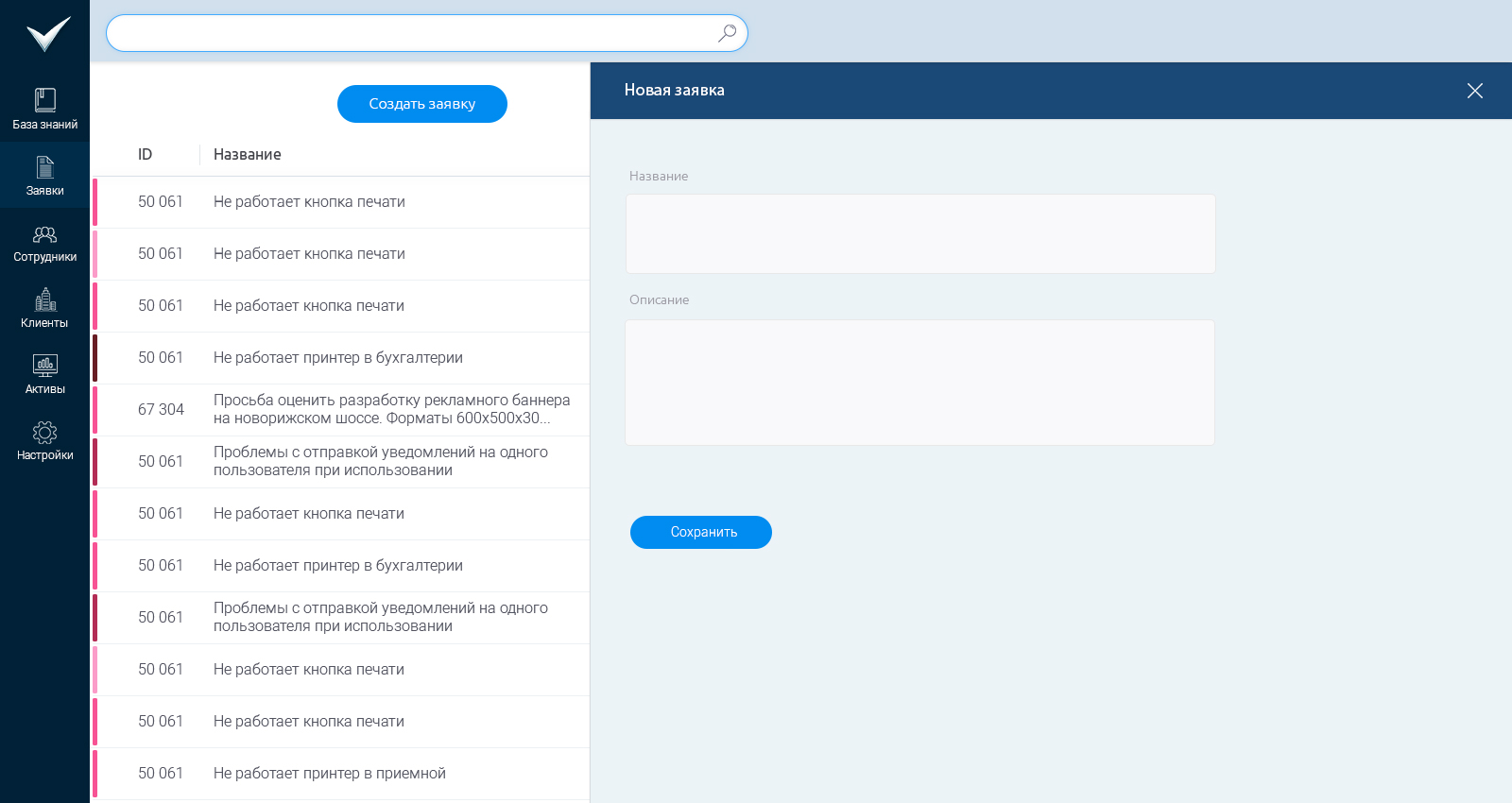
Рисунок . Список заявок

В PSD это слои:



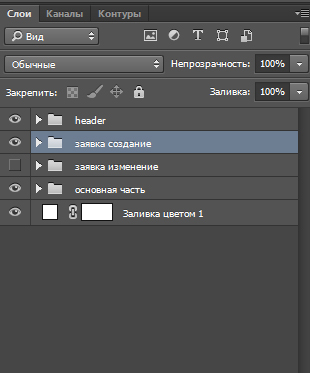
Список без постраничного листания, отображать все записи сразу с вертикальным скролом, если записей много.  
(Разумеется, все данные необходимо получать из API, см. ниже)

* Клик на кнопку «Создать заявку» должен открывать форму создания заявки, частично перекрывая список.   
  Левые и правые колонки должны иметь независимый скрол, т.е. карточка заявки не должна уезжать, когда скролируется список заявок.  
    
  Пользователь должен иметь возможность заполнить два поля (Название и Описание как на макете) и нажать кнопку «Сохранить», рисунок 2.



Рисунок

В PSD это слои:



При этом должна быть создана заявка (через запрос к API) и карточка заявки автоматически должна перейти в режим редактирования данной заявки.

* Клик на заявку в списке должен приводить к открытию справа соответствующей карточки заявки на редактирование. При редактировании Заявки достаточно реализовать:
  + Добавление комментария
  + Изменение статуса. (В правом верхнем углу «В работе». Самостоятельно подобрать наиболее подходящий контрол для изменения, так чтобы пользователь понимал, что он может изменить статус у заявки. Выбор значения из справочника статусов (запрос к API на список статусов)
  + Изменение исполнителя. Любая реализация. Выбор значения из справочника пользователей (API Endpoint http://intravision-task.test01.intravision.ru/api/ {tenantguid}/Users).

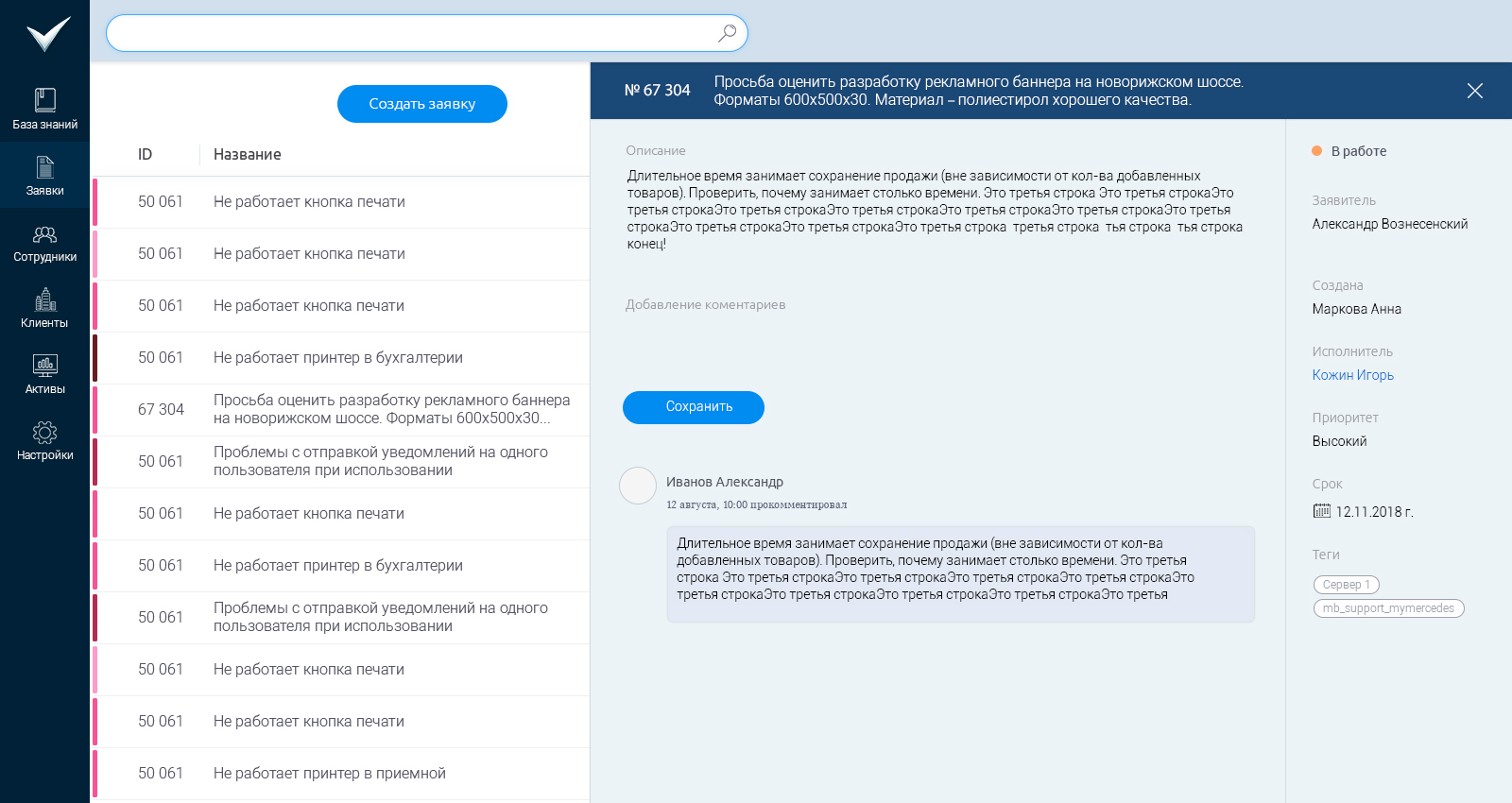
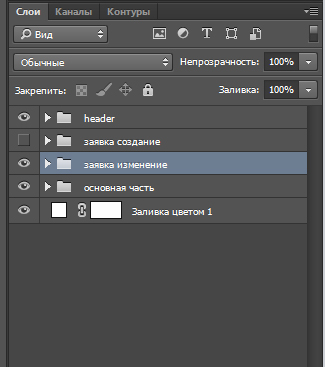


Рисунок . Редактирование заявки

В PSD это слои:



Аналогично созданию заявки левая и правая область должны скролироваться независимо.

* Иконки левого меню «База знаний», «Пользователи», «Активы», «Настройки» должны открывать соответствующие разделы. Каждый такой раздел это пустая страница с заголовком.
* «Поиск ответа по базе знаний». Просто сверстанное поле, без функциональности поиска.

## Требования

Разработка с использованием React. Взаимодействие с серверной частью с использованием Web API (Redux, Context API на Ваш выбор). Ознакомиться с документацией и проверить работу методов API можно по ссылке <http://intravision-task.test01.intravision.ru/swagger/index.html>

Для выполнения тестового задания, требуется получить tenantguid (необходимо, чтобы с одним API могло работать сразу несколько человек без пересечения), для этого необходимо перейти по ссылке <http://intravision-task.test01.intravision.ru/api/Tenants>. Ответ на запрос будет содержать уникальный guid. Полученный guid нужно использовать во всех запросах к API.

Если у вас есть вопросы по работе API или вы считаете, что API неправильно работает, смело пишите – минусом это считаться не будет.

К проекту приложите файл readme.txt с инструкцией установки и запуска проекта.

## Результаты тестового задания

* Результаты выслать в архиве. Имя архива “ФамилияИмя\_ТестовоеЗадание.zip”
* Необходимо в архив с заданием также положить файл readme.txt с кратким описанием того что сделали, того что не сделали. Всё что посчитаете нужным
* В архив стоит включать только исходники проекта
* В архив не стоит включать папку “node\_modules”

## Подсказки

1. Справочник приоритетов (API Endpoint: GET http://intravision-task.test01.intravision.ru/api/{tenantguid}/Priorities) и статусов(API Endpoint: GET http://intravision-task.test01.intravision.ru/api/{tenantguid}/Statuses) мы рекомендуем загружать сразу при открытии страницы, а не «дергать» в момент работы пользователя с интерфейсом.
2. При изменении заявки (не путать с созданием) метод API (EndPoint: PUT [http://intravision-task.test01.intravision.ru/api/{tenantguid}/Tasks](http://intravision-task.test01.intravision.ru/api/%7btenantguid%7d/Tasks)) используется как для изменения информации по заявке, так и для добавления комментариев. В запросе достаточно передавать только те данные, которые изменяются.
3. Для того, чтобы увидеть список всех заявок (для отладки во время разработки), можете использовать метод API (API Endpoint: GET [http://intravision-task.test01.intravision.ru/odata/tasks?tenantguid={tenantguid}](http://intravision-task.test01.intravision.ru/odata/tasks?tenantguid=%7btenantguid%7d))

Удачи в разработке!