

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное бюджетное образовательное учреждение
Высшего образования
«Московский политехнический университет»

Лабораторная работа
по дисциплине: «Программная инженерия»
на тему: Анализ существующих подобных программных продуктов

Выполнил студент группы
Рубенов Александр Аликович
181-322

Проверила преподаватель:
Будылина Евгения Александровна

Москва, 2020

1. Выбор и описание программного продукта

Firebase — американская компания, поставщик облачных услуг, основанная в 2011 году Эндрю Ли и Джеймсом Тэмплином, и поглощённая в 2014 году корпорацией Google.

Основной сервис — облачная СУБД класса NoSQL, позволяющая разработчикам приложений хранить и синхронизировать данные между несколькими клиентами. Поддержаны особенности интеграции с приложениями под операционные системы Android и iOS, реализовано API для приложений на JavaScript, Java, Objective-C и Node.js, также возможно работать напрямую с базой данных в стиле REST из ряда JavaScript-фреймворков, включая AngularJS, React, Vue.js, Ember.js и Backbone.js. Предусмотрено API для шифрования данных.

Сайт компании - <https://firebase.google.com/>

Текущая версия продукта – Firebase log 2.4.8 (Разная для каждой платформы)

2. Требования к проектируемой информационной системе (Функциональное назначение)

Firebase предоставляет следующие функциональные возможности:

- **Авторизация** – Поддержка регистрации и авторизации пользователей, восстановление паролей и доступа к ЛК.
- **Создание мультибазы** – Поддержка нескольких БД в одной, для учета заявок, назначение персонала.
- **Создание автоматической базы под проект** – После интеграции продукта пользователя и СМР, автоматически открывается и загружается БД пользователя.
- **Мобильное обслуживание** - Обслуживание БД с мобильного носителя
- **Учет клиентов и персонала** – ведение единой базы, с авто-изменениями

3. Требования программного обеспечения и возможности интеграции

Доступ к программе бывает нескольких видов и зависит от языка разработки проекта:

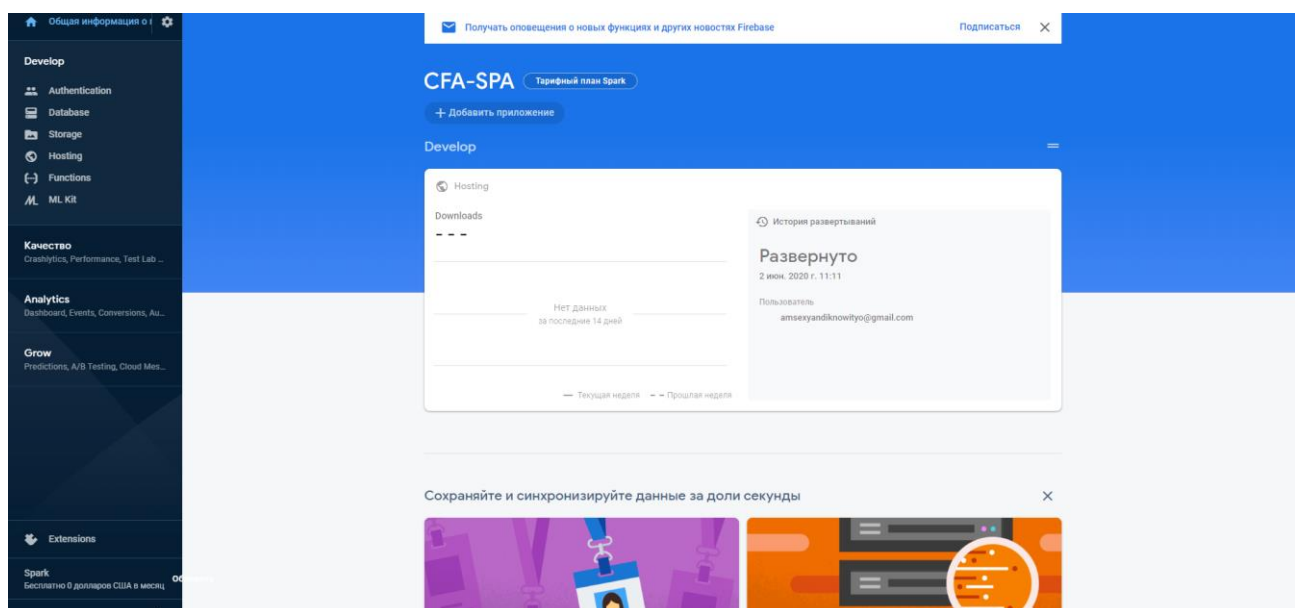
- Можно подключить через прямой REST файл, без скачивания пакета данных firebase.
- Можно же скачать пакет данных и подключить необходимые ссылки в проект.

Все работы – производит менеджер IT отдела.

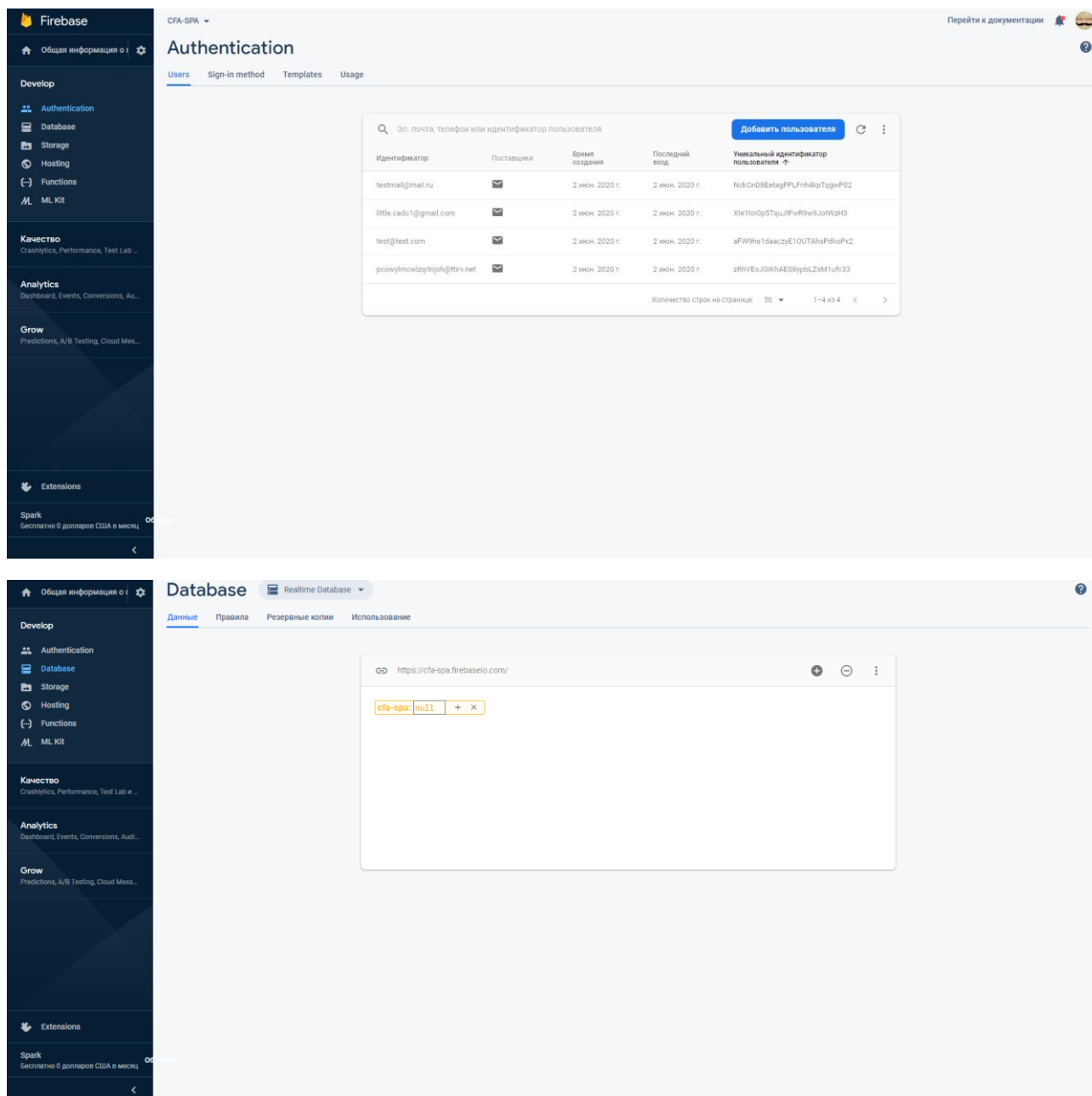
Платформа интегрируется со всеми существующими приложениями, поддерживающие REST технологии (98% сервисов)

4. Организация интерфейса

Интерфейс программы неоднократно перерабатывался, чтобы стать удобным для всего круга пользователей. Он представляет собой минималистичную и понятную пользователю палитру. Освоение занимает меньше минуты, очень удобная и понятная навигация.



Таким образом выглядит консоль разработчика, в которой можно перейти в различные вкладки, а настройки различных БД, просмотр пользователей, хранилище и тд.



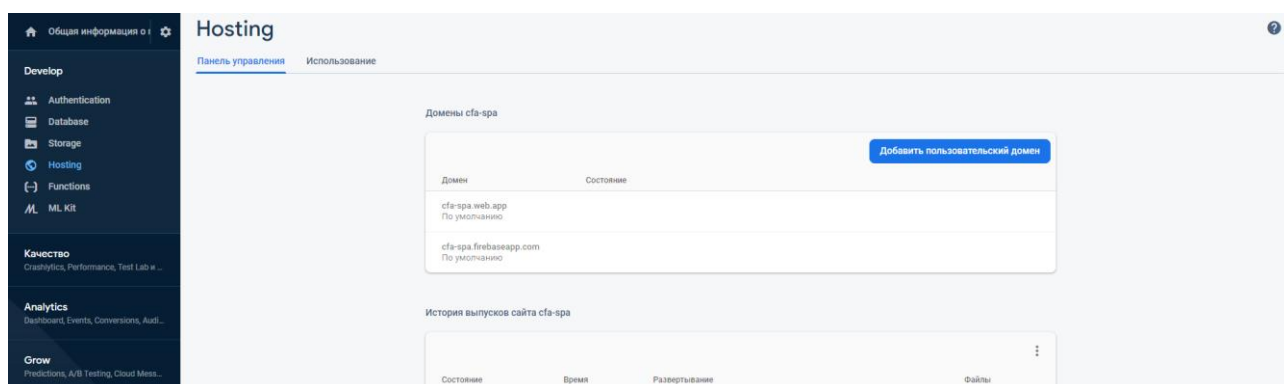
Таким образом выглядит панель БД. (Сразу хотелось бы отметить, что это не CMR, в которой показаны заявки и прочее, это хранилище данных, которое взаимодействует с визуальным сервером, написания данных). В таких ячейка хранятся объекты памяти приложения, которые в дальнейшем визуализируются.

Информации можно писать много – сервис многофункционален и гибок, поэтому он находится в ТОП 1 рейтинга по новым REST БД.

5. Структура меню

Пункт меню	Описание
Авторизация	Список пользователей с различными правами. Данные для ЛК каждого пользователя по защищенному протоколу. Информация с возможностью восстановления и регистрации пользователя.
БД	Список БД, доступных для проекта. Регистрация новых данных, их изменение и удаление, как локально, так и удаленно.
Хранилище	Все данные пользователей в одной защищенной коробке
Хостинг	Режим бесплатного хостинга, с возможностью добавления приложения пользователя, как на стадии разработки, так и на финальной стадии.
Аналитика	Добавления аналитики приложения, SMM/SEO серверные настройки и прочее
Расширения	Персональные наработки и скрипты администратора
Функции	Полный перечень возможного расширенного функционала

Еще раз подчеркну, что этот прекрасный ресурс служит в роли БД, к визуальному веб интерфейсу, с помощью которого мы добавляем заявки, отслеживаем их, назначаем задачи, и проводим debug систему.



6. Работа с данными

Пользователь оставляет заявку на нашем сайте, которая сразу попадает в нашу систему *Firebase* по хранению заявок в БД. На своем сервисе, у нас визуально отображается, что пришла новая заявка.

После этого, администратор, с аналогичной ролью на сервисе, назначает исполнителя задачи, данные посылаются снова в *Firebase*, где изменяется статус и отдельным параметром присваивается исполнитель, в визуальном интерфейсе так же происходят изменения.

В БД клиентов добавляется новый клиент, с соответствующим брифом с клиентами данными. В БД сотрудников изменяется задача, выбранного исполнителя. Все изменения переносятся на виртуальный интерфейс.

Создается БД клиента, и автоматически деплоится на хостинг *Firebase*.

По завершению работы производятся полностью обратные операции, и готовая БД клиента ставится ему на ресурс и вся права также передаются клиенту.

7. Вывод

Проанализировав данное программное обеспечение, мы выбрали следующие удачные решения, при реализации ПИ и серверной части:

1. Мульти БД
2. Автоматический хостинг
3. Возможность передать существующую БД, после деплоя – клиенту
4. Полная гибкость
5. Наличие мобильного администрирования
6. Live режим
7. Поддержка всех существующих языков и интерфейсов для интеграции