

УДК 004.42

ОБОСНОВАНИЕ ВЫБОРА FIREBASE КАК СУБД СОЦИАЛЬНОЙ СЕТИ С ГЕОПОЗИЦИНИРОВАНИЕМ И ДОПОЛНЕННОЙ РЕАЛЬНОСТЬЮ

Владимир Алексеевич Раевский,

ФГБОУ ВО «Калужский государственный университет им. К.Э. Циолковского», доцент кафедры «Информатики и информационных технологий», г. Калуга, var-77@mail.ru.

Дмитрий Сергеевич Аверин,

ФГБОУ ВО «Калужский государственный университет им. К.Э. Циолковского», магистрант кафедры «Информатики и информационных технологий», Калуга, averinds@studklg.ru.

Аннотация

В статье обосновывается актуальность разработки социальной сети с геолокацией пользователей и применением технологии дополненной реальности, одним из элементов которой является система управления базой данных. Проводится сравнительный анализ СУБД Firebase и SQL в реализации MS SQL Server. Рассматриваются достоинства и недостатки Firebase, обосновывается окончательный выбор СУБД для разработки упомянутой социальной сети.

Ключевые слова: Firebase, Android, СУБД, приложение, программная разработка.

JUSTIFICATION FOR CHOOSING A FIREBASE AS A SOCIAL NETWORK DBMS WITH GEOPOSITIONING AND AUGMENTED REALITY

Vladimir A. Raevsky,

Kaluga State University named after K.E. Tsiolkovski, Associate Professor of the Department of Informatics and Information Technologies, Kaluga city, var-77@mail.ru.

Dmitriy S. Averin,

Kaluga State University named after K.E. Tsiolkovski, undergraduate of the Department of Informatics and information technologies, Kaluga city, averinds@studklg.ru

ABSTRACT

The article substantiates the relevance of developing a social network with user geolocation and the use of augmented reality technology, one of the elements of which is a database management system. A comparative analysis of the Firebase and SQL DBMS in the implementation of MS SQL Server is carried out. The advantages and disadvantages of Firebase are considered, the final choice of a DBMS for the development of the mentioned social network is substantiated.

Keywords: Firebase, Android, DBMS, application, software development.

В настоящее время разработка социальных сетей направлена, в том числе, на применение технологии AR (Augmented Reality, дополненная реальность). Интеграция технологий дополненной реальности и геолокации в социальную сеть позволяет пользователям получать возможность определения желаемых маркеров в реальном мире и перемещения к ним, легкой визуализации местоположения своих контактов и быстрого выбора их в «переполненном» пространстве. Таким образом, разработка социальной сети с элементами дополненной реальности является актуальной проблемой, при этом одним из элементов разрабатываемой социальной сети является система управления базой данных (СУБД).

Для реализации задачи могут использоваться как обычные СУБД SQL(RDBMS)), так и платформы Google Firebase, Amazon AWS, Microsoft Azure и др., включающие в себя сервисы облачной базы данных, уведомлений, хранения статических и динамических данных, аналитики и др. позволяющие упростить создание и развертывание полнофункциональных систем [1].

Было выполнено сравнение и обоснование выбора СУБД для разрабатываемой социальной сети.

Firebase – это платформа для разработки мобильных устройств и приложений, которая была разработана Firebase Inc, но позже была приобретена Google [2 – 4]. Основным сервисом Firebase является облачная база данных NoSQL, которая не использует таблицы, не хранит свои данные (по умолчанию) локально на устройстве, а содержит их в облаке (ярлык «Real-time») [5].

Firebase предлагает разработчикам ряд услуг, которые облегчают разработку приложений [6 – 8]. Некоторые из наиболее популярных сервисов включают в себя:

1. Google Analytics. Анализирует атрибуцию и поведение пользователей и предоставляет единую панель мониторинга для ее визуализации и управления.
2. Аутентификация. Предоставляет простой способ авторизации пользователей с помощью нескольких методов, включая электронную почту и пароль, а также сторонние сервисы, такие как Google, Facebook и т. д.
3. Облачные функции – пользовательский серверный код, который запускается некоторыми заданными событиями. Выполняется и масштабируется автоматически без необходимости управления собственными серверами.
4. База данных в реальном времени. Хранит и синхронизирует данные между различными абонентами и устройствами в режиме реального времени с помощью базы данных NoSQL, обновленные данные синхронизируются за миллисекунды независимо от подключения к сети.
5. Облачные сообщения. Отправление сообщений и уведомлений как пользователям, так и отдельным устройствам или группам устройств, на разных платформах, Android, iOS и т. д.

В таблице 1 показано сравнение Firebase и SQL, которое основано на таких значимых параметрах, как хранение данных, модель данных, гибкость с точки зрения схемы и т. д.

Таблица 1. Сравнение Firebase и SQL

Наименование параметра	Firebase	SQL(RDBMS)
Хранение данных	Хранится в виде дерева JSON	Хранится в Реляционной модели в виде строк и столбцов (таблиц)
Модель данных	Хранится в виде дерева JSON	Хранится в виде таблицы
Вид данных	Данные, которые не имеют определенного типа или структуры	Данные, тип которых известен заранее
Гибкость схемы	Динамическая схема, данные могут быть добавлены, обновлены или удалены в любое время	Изменение приведет к временному отключению БД
Наличие SQL	Отсутствует	Присутствует
Техника	Синхронизация данных	Fire Query

Как видно из таблицы 1, хранение и данных и их модель в виде дерева JSON, а также динамическая схема позволяют в первом приближении рекомендовать именно Firebase как СУБД для социальной сети с геопозиционированием и технологией AR.

Рассмотрим более подробно преимущества и недостатки Firebase.

Преимущества Firebase:

1. Поддерживает высококачественные базы данных, которые можно использовать с приложениями, которые можно масштабировать в зависимости от размера.
2. Полностью безопасное решение для управления приложениями, позволяющее разработчикам легко получать доступ к данным через консоль Firebase. Автономный доступ и «плавное» обновление данных обеспечивают синхронную работу нескольких баз данных в режиме реального времени.
3. В Firebase встроен ряд сервисов (продуктов), обеспечивающих эффективную работу приложений. Например, благодаря сервисам, использующим интегрированные облачные функции, проект можно хранить в облаке, что позволяет разрабатывать «бессерверные» приложения.
4. Доступный пользовательский интерфейс. В большинстве случаев службы разработки Firebase требуют минимальных знаний и информации о языках программирования и упрощают интеграцию через пользовательский интерфейс. Не рассматривая это как недостаток гибкости, проект может избежать использования сложных конфигураций.
5. Компетенции статического хостинга. Firebase hosting предоставляет встроенную сеть доставки контента (CDN) с облачной платформой Google. В

частности, этот CDN представляет собой систему распределенных серверов, которая способствует более быстрой доставке контента по всему миру. С Firebase можно использовать его возможности статического хостинга для разработки продуктивных одностраничных и веб-приложений [9].

Недостатки Firebase:

1. Некоторые ограничения базы данных реального времени. Например, ограниченные возможности запросов являются одной из наиболее доминирующих проблем; постоянно увеличивающийся объем информации в файле JSON создает некоторые трудности при вводе сложных запросов. Из-за структуры «база данных как единый файл» реализация связей между элементами данных невозможна.
2. Firebase как платформа для разработки приложений не имеет «привязки к поставщику». Это одна из основных проблем, которая возникает, когда необходимо использовать серверную службу Firebase. Более того, поскольку это технологическое ограничение не предлагает инструментов миграции для переключения данных на другую платформу, что оказывается серьезным недостатком.
3. Ограниченная поддержка iOS, так как Firebase ориентирована более на мобильную платформу Android [10].

Исходя из вышесказанного, окончательно в качестве СУБД разрабатываемой социальной сети выбрана платформа Firebase. Для большинства разработчиков она позволяет не сосредотачиваться на серверном коде, а реализовывать непосредственно приложение СУБД; содержит необходимые функции, такие как отправка сообщения пользователям, хранение данных, масштабируемая база данных с возможностью выборки и адаптивной аутентификации.

Список литературы:

1. What Can I Do With Cloud Functions [Электронный ресурс] // Firebase. – Режим доступа: <https://firebase.google.com/docs/functions/use-cases> (дата обращения: 11.06.2022 г.).
2. Буй М. Д. Google Genomics API / М. Д. Буй, В. К. Хо, Ч. Т. Лыу // Синергия наук. – 2017. – № 8. – С. 275-282.
3. Буй М. Д. Анализ современных технологий для создания мобильных приложений / М. Д. Буй, Т. Е. Войтюк // Альманах науч. работ молодых ученых ун-та ИТМО. Т. 1. – СПб., 2016. – С. 188-191.
4. Ахметов А. К. Операционная система Android: история создания и развития. Разработка приложений для платформы Android // Скиф. Вопр. студенческой науки. – 2017. – № 9. – С. 108-115.
5. Oryschak C. Launching Cloud Functions for Firebase v1.0 [Электронный ресурс] // Firebase. – Режим доступа: <https://firebase.googleblog.com/2018/04/launching-cloudfunctions-for-firebase-1-0.html> (дата обращения: 12.06.2022 г.).

6. Android Developers [Электронный ресурс] : [официальный сайт]. – Режим доступа: <https://developer.android.com/distribute/best-practices/develop/build-withfirebase.html?hl=ru#key-benefits>. – Загл. с экрана (дата обращения: 10.06.2022 г.).
7. Firebase Realtime Database [Электронный ресурс] // Firebase. – Режим доступа: <https://firebase.google.com/docs/database> (дата обращения: 10.06.2022 г.).
8. Лэйдр В. Х. How to convert SQL structures to NoSQL structures of Firebase // Синергия наук. – 2017. – № 8. – С. 370-375.
9. Белунина Е. П. Идеология облачных вычислений [Электронный ресурс] / Е. П. Белунина, Е. Д. Маштакова, Л. Г. Симульман // Школа университетской науки: парадигма развития. – 2016. – № 3-4 (21-22). – С. 265-268.
10. Варфоломеева Т. Н. Современное состояние и перспективы развития облачных технологий в России / Т. Н. Варфоломеева, А. Р. Габитова // Лучшая науч.-исслед. работа 2017 : сб. ст. VIII международ. науч.-практ. конкурса. – Пенза, 2017. – С. 15-18.

References:

1. What Can I Do with Cloud Functions [Electronic resource] // Firebase. – Access mode: <https://firebase.google.com/docs/functions/use-cases>. (date of the application: 11.06.2022 г.).
2. Buj M. D. GoogleGenomicsAPI / M. D. Buj, V. K. Ho, CH. T. Lyu // Sinergiya nauk. – 2017. – № 8. – P. 275-282. (in Russian).
3. Buj M. D. Analiz sovremennykh tekhnologij dlya sozdaniya mobil'nyh prilozhenij / M. D. Buj, T. E. Vojtyuk // Al'manah nauch. rabot molodyh uchenyh un-ta ИТМО. V. 1. – St. Petersburg., 2016. – P. 188-191. (in Russian).
4. Ahmetov A. K. Operacionnaya sistema Android: istoriya sozdaniya i razvitiya. Razrabotka prilozhenij dlya platformy Android // Skif. Vopr. studentcheskoj nauki.– 2017. – № 9. – P. 108-115. (in Russian).
5. Oryschak C. Launching Cloud Functions for Firebase v1.0 [Electronic resource] // Firebase. – Access mode: <https://firebase.googleblog.com/2018/04/launching-cloudfunctions-for-firebase-1-0.html>. (date of the application: 12.06.2022 г.).
6. Android Developers [Electronic resource] : [official site]. – Access mode: <https://developer.android.com/distribute/best-practices/develop/build-withfirebase.html?hl=ru#key-benefits>. (date of the application: 10.06.2022 г.).
7. Firebase Realtime Database [Electronic resource] // Firebase. – Режим доступа: <https://firebase.google.com/docs/database> (date of the application: 10.06.2022 г.).
8. Leidr V. H. How to convert SQL structures to NoSQL structures of Firebase // Sinergiya nauk. – 2017. – № 8. – P. 370-375. (in Russian).
9. Belunina E. P. Ideologiya oblachnyh vychislenij [Elektronnyj resurs] / E. P. Belunina, E. D. Mashtakova, L. G. Simul'man // SHkola universitetskoj nauki: paradigma razvitiya. – 2016. – № 3-4 (21-22). – С. 265-268. (in Russian).
10. Varfolomeeva T. N. Sovremennoe sostoyanie i perspektivy razvitiya oblachnyh tekhnologij v Rossii / T. N. Varfolomeeva, A. R. Gabitova // Luchshaya nauch.-issled. rabota 2017 : sb. st. VIII mezhdunarod. nauch.-prakt. konkursa. – Penza, 2017. – P. 15-18. (in Russian).