Academic Journal of Nawroz University (AJNU), Vol.11, No.3, 2022

https://doi.org/10.25007/ajnu.v11n3a1480

Firebase Efficiency in CSV Data Exchange Through PHP-Based Websites

Ridwan Boya Marqas[1], Saman M. Almufti [2], Renas Rajab Asaad[3]
[1,2,3]Department of Computer Science, Nawroz University, Kurdistan Region – Iraq

ABSTRACT
A database is a collection of data that may be organized into two distinct forms: relational databases, which use SQL (structure query language), and distributed databases, which use NoSQL (non-relational SQL). Both types of databases are referred to as databases. The amount of information around the globe is increasing at an exponential rate, leading to the development of data as the world gets more technologically advanced and computerized. In the vast majority of studies, the NoSQL database is simply referred to by its acronym, "NoSQL." In this investigation, data is transferred from a website written in PHP to a CSV file by way of a NoSQL database known as Firebase. To import and export the experimental data, it was necessary to use two different CSV files, each of which included 1,000 and 4,997 records, respectively. Exchanging CSV files with a website that was built using PHP was the method that this study used to test the performance of Firebase.

KEYWORDS: Database, Firebase, Website, PHP, NoSQL.

1. INTRODUCTION

The term "database" is mostly used to refer to a collection of data that has been structured, saved, and accessible via a computer system (Ullman 2007). Today's digital systems are susceptible to data with huge dimensions because of the extent, diversity, and complexity of the data universe. [citation needed] In addition, enormous amounts of data must be saved, managed, or analyzed using cloud and social media storage and management systems (Ramzan and Bajwa 2018). NoSQL databases are used for enterprise and open-source database management. These databases store vast amounts of data across a network of computers. The name "Not Just SQL" was given to the NoSQL database management system to dispel the widespread belief that SQL cannot be contained (Ganesh Chandra 2015).

NoSQL makes it possible to scale horizontally; hence, its many implementations are always kept on distinct servers. The column-based NoSQL database stores data in a single huge table, as opposed to the relational database, which stores data in numerous tables. This is done to make autoscaling easier (Lee and Zheng 2015). Firebase is a NoSQL database that is hosted on the cloud and runs in that environment. Even when the local cache network is off, it is still feasible to synchronize all the devices that are linked together. This database is driven by events, and in comparison, to ordinary SQL databases, it operates considerably differently (Moroney and Moroney 2017a). PHP was initially developed by Rasmus Lerdorf, a Canadian of Danish ancestry, and the most recent version, PHP 7, incorporates all the most recent enhancements (Jentsch 1997).

This article looks at how well Firebase performs while importing and exporting CSV files with websites that are based on the PHP programming language.

2. MATERIALS AND METHODS

A. Programming Hypertext Protocol (PHP)

Web pages are often built using a dynamic programming language called PHP, which may be

found on a broad variety of websites and web applications that are constantly updated (Mon et al. 2019). To access and save data on a variety of platforms, the PHP code may be modified to operate with other web scripting languages. Comma Separated Value (CSV) files may be read and written using functions in PHP (Mon et al. 2019).

Large volumes of data are frequently sent across disconnected applications using CSV files. Commas are commonly used to separate spreadsheet fields in CSV files, whereas system end-of-line characters are commonly used to separate CSV records in Microsoft Excel or Text pad (Hapeez, Yassin, and Hamzah 2010).

2.1 Non-relational database (NoSQL)

Non-relational databases, such as NoSQL, do not have a defined structure and may be accessed using a simple query language. Data may be distributed and reproduced in a less controlled environment using NoSQL's huge database. For the foreseeable future, this database will maintain the ability to store data independently of any other databases. NoSQL databases are organized hierarchically. It's also capable of handling large volumes of data at high speeds. The structure of a NoSQL database is horizontal. NoSQL databases like Cassandra and HBase are among the few that may be used in a NoSQL application.

In general, there are numerous Relational databases and non-Relational databases. SQL databases include MySQL, Oracle, Microsoft SQL Server, PostgreSQL, and DB2, whereas NoSQL databases include Redis, Amazon DynamoDB, Cassandra, Scylla, HBase, Firebase, MongoDB, Couchbase, Neo4j, Datastax Enterprise Graph, Elasticsearch, Splunk, and Solr. See Fig. 1.

Fig. 1. Classes of SQL and NoSQL

This table compares several aspects of SQL and NoSQL Databases, such as their scalability and pricing as well as their capacity to hold a large quantity of data and how quickly they can be accessed.

3. PROPOSED METHOD

This section describes the proposed technique for reading CSV files and importing data into the firebase database, as well as exporting data from the firebase service. The execution time was calculated to assess the effectiveness of importing and exporting PHP-based website files using firebase Information.

TABLE 1
Comparison of SQL and NoSQL criteria

Criteria	Relational Database (SQL)	NoSQL
diversity	Open and closed source	Open source
Scalability	Upgrade a single server with devices	Using standard servers scale horizontally
price	Costly data access solution	inexpensive than open source and cheaper update
Amount of data	Limited	Vast data hold
accessibility	Affect by single fail	Unaffected by one point of failure that's distributed
Execution time	Long process time	Short process time
Complexity	Complex data creation	Less complex data creation
implementation	Small improvement occurs	Each stage own improvement occurs
uniformity	Structured	Unstructured
Security	Strong Security	Security not included is related to other parts

3.1 PHP & CSV

PHP needs the following functions for reading and writing data to and from CSV files:

I. FOpen operation

This function is used to open a CSV file

using open (file, mode), where the file represents the target file and mode represents the access required for reading or writing in the CSV file.

II. Fgetcsv function

This function is used to read data from a CSV file by iteratively parsing an open file line by line and checking for data fields. fgetcsv (file, length, separator), where the file is the target file, length is the maximum length of a CSV row, and separator is a comma to separate CSV data.

III. Fputcsv function

This function is used to write data to a CSV file fputcsv (file, fields), where the file is the destination file and fields are the data array.

IV. Fclose Method

This function is used to open the CSV file fclose (file), where the file represents the file to be opened.

The features of Firebase are broken down into several categories and described in Table 2.

TABLE 2
Characteristics of firebase

Criteria	FIREBASE
Type	Cloud hosted Realtime
Database Model	Document Store
Develop by	Google company
Release	2012
Commercial	Yes
Cloud based	Yes
Server OS	Hosted
Scheme of Data	Free schema
XML support	No
SQL	No
Access methods and API's	Android
Support program language	iOS
Server-side scripts	JavaScript API
Triggers	RESTful HTTP API
Consistency	Java
Foreign keys	JavaScript
Integrity	Objective-C
Authentication	Functionality are limited with rules

3.2 PHP & FIREBASE

Firebase is a cloud-based, real-time database developed for mobile and web apps, although it cannot be used directly with PHP to create websites. Firebase stores data as JSON; hence, the Composer dependency manager, which provides a standard structure for managing PHP and library dependencies, is required for linking PHP with Firebase.

I. Getreference function

This function is used to retrieve and push values from the source database.

getreference (DB)

where DB is the Database being referenced.

II. Push functionality

This method inserts a list of data records into the Firebase database. When adding a node to a list of data, the Firebase database generates a new unique key.

Push(Data)

Data represents the list of data to be uploaded into Database.

III. The getValue method

retrieves data from the Firebase database.

getValue()

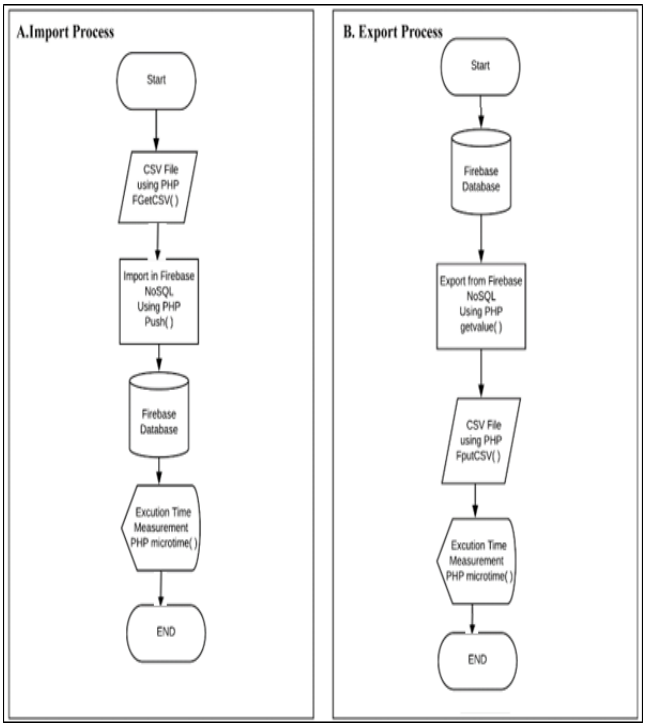

Fig. 2. process of import & export.

Figure 2 depicts the data import and export processes for both the firebase databases.

4. EXPERIMENT & RESULT

This section demonstrates the results of an experiment including the import and export of data from the firebase databases, with download speeds of 34.65 Mbps and upload speeds of 36.06 Mbps.

In the experiment, two hospital-related CSV files were employed; the first CSV file included 1000 entries, while the second CSV file contained 4997 records; both files had 11 columns, as seen in (figure 3).

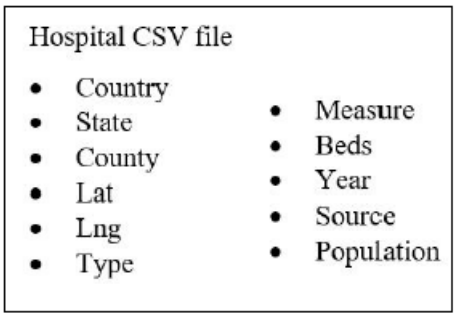

Fig. 3. columns name of CSV file

(Figure 4) depicts the full system connection beginning with the connection from the computer to the 000webhost server. The server hosts the website that connects to the distinct databases listed below:

• Firebase: in Google server Using NoSQL.

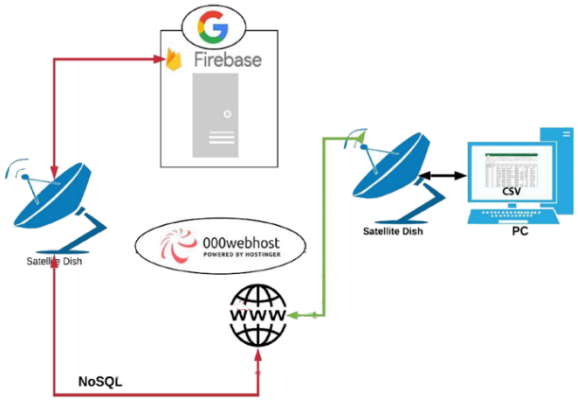

Fig. 4. Connection System

A connection system in (figure 4) demonstrates the connection of the system used that import and export process occurred through it which the connection distributes between firebase and CSV file by using 000webhost hosting server.

In the Firebase online database, a table named hospital was created according to the CSV structure depicted in (figure 5).

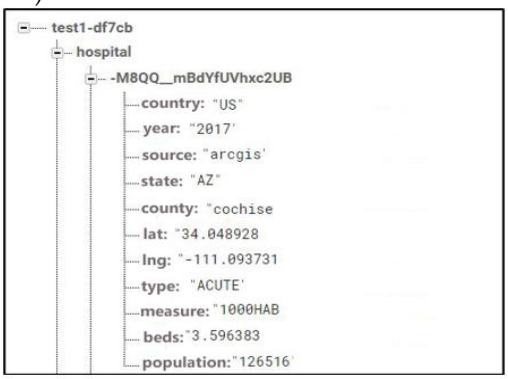

Fig. 5. demonstrates the hospital table database structure online Firebase

Table 3 depicts the execution times for importing and exporting data in the firebase database.

TABLE 3
duration of Firebase's execution

process	Record No.	Firebase
import	1000	138.649094 seconds
	4997	735.564 seconds
export	1000	0.495048046 seconds
	4997	0.774774 seconds

Table 3 displays the import and export execution times for 1000 and 4997 CSV entries in firebase.

(Figure 6) depicts the import process execution time for 1000 and 4,997 CSV records into Firebase, while (Figure 7) depicts the export process execution time for 1000 and 4,997 CSV records in the Firebase database.

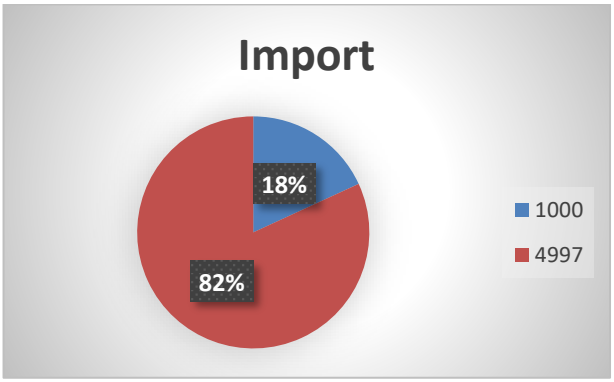

Fig. 6. demonstrates the import process for Firebase

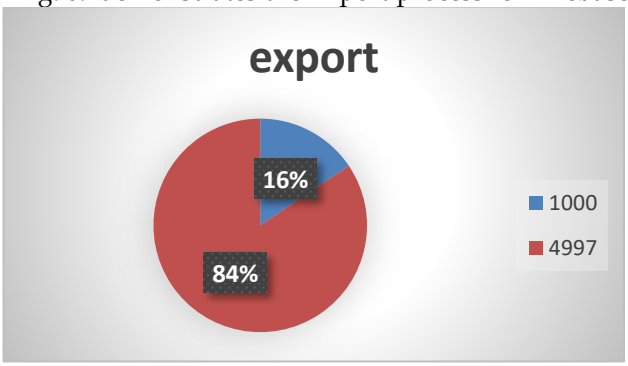

Fig. 7. demonstrates the export process for Firebase

The import and export process show the percentage for 1000 and 4997 CSV record which for import one use 18% for 1000 and 82% for 4997, while the export process show 16% for 1000 and 84% for 4997 CSV record.

5. CONCLUSION AND RECOMMENDATIONS

This study presents a time performance assessment for the process of data transferring (import and export) between CSV files and Firebase online databases with a PHP-based website.

The experimental results for two CSV files containing 1000 and 4997 records indicate that the data exporting process for Firebase in a PHP-based website is faster than the importing process. This is because, during the import process, the connection between the website and Firebase is indirect and requires more time to reach the database on an external server, whereas the export process directly reaches the database and retrieves the data.

REFERENCES

Binani, Sneha, Ajinkya Gutti, and Shivam Upadhyay. 2016. "SQL vs. NoSQL vs. NewSQL- A Comparative Study." 6(1):43–46.

DB-Engines.com. 2020. "Firebase Realtime Database vs. Sqlite Comparison." Retrieved June 2, 2020 (https://db-engines.com/en/system/Firebase+Realtime+Database%3BSQLite).

Ganesh Chandra, Deka. 2015. "BASE Analysis of NoSQL Database." Future Generation Computer Systems 52:13–21.

Hapeez, Mohammad Shukri, Mohd Ihsan Mohd Yassin, and Mustafar Kamal Hamzah. 2010. "Storage of Online HPLC Data for Pharmaceutical Research Applications Using XML Database." Proceedings of the 2010 5th IEEE Conference on Industrial Electronics and Applications, ICIEA 2010 1605–9.

Jentsch, Birgit. 1997. Gender Politics and Post-Communism: Reflections from Eastern Europe and the Former Soviet Union. Vol. 3.

Lee, Chao Hsien and Yu Lin Zheng. 2015. "Automatic SQL-to-NoSQL Schema Transformation over the MySQL and HBase Databases." 2015 IEEE International Conference on Consumer Electronics - Taiwan, ICCE-TW 2015 426–27.

Mon, Cho Thet, Su Su Hlaing, Mie Mie Tin, Mie Mie Khin, Tin Moh Moh Lwin, and Khin Mar Myo. 2019. "Code Readability Metric for PHP." 2019 IEEE 8th Global Conference on Consumer Electronics, GCCE 2019 929–30.

Moroney, Laurence and Laurence Moroney. 2017a. "An Introduction to Firebase." The Definitive Guide to Firebase 1–24.

Moroney, Laurence and Laurence Moroney. 2017b. "The Firebase Realtime Database." The Definitive Guide to Firebase 51–71.

Ramzan, Shabana and Imran Sarwar Bajwa. 2018. "SS Symmetry An Intelligent Approach for Handling Complexity by Migrating from Conventional Databases to Big Data."

Stephens, Jon and Chad Russell. 2004. Beginning MySQL Database Design and Optimization. Ullman, J. D. 2007. "A First Course in Database Systems." Pearson Education India. Wang, K. C. 2018. Systems Programming in Unix/Linux.