



Redis

Reviews, tips, and advice from real users



Powered by  PeerSpot

Contents

Product Recap..... 3 - 4

Valuable Features..... 5 - 12

Other Solutions Considered..... 13 - 15

ROI..... 16 - 19

Use Case..... 20 - 23

Setup..... 24 - 27

Customer Service and Support..... 28 - 29

Other Advice..... 30 - 32

Trends..... 33 - 34

About PeerSpot..... 35 - 36

Product Recap



Redis

Redis Recap

Redis offers high-speed, in-memory storage, renowned for real-time performance. It supports quick data retrieval and is used commonly in applications like analytics and gaming.

Renowned for real-time performance, Redis delivers high-speed in-memory storage, making it a favorite for applications needing quick data retrieval. Its diverse data structures and caching capabilities support a broad array of use cases, including analytics and gaming. Redis ensures robust scalability with master-slave replication and clustering, while its publish/subscribe pattern renders it reliable for event-driven applications. The solution integrates smoothly with existing systems, minimizing performance tuning needs. Although documentation on scalability and security could be improved, Redis remains cost-effective and stable, commonly utilized in cloud environments. Enhancing integration with cloud services like AWS and Google Cloud and refining GUI may improve usability.

What are the key features of Redis?

- **High Speed:** In-memory data storage for swift access.
- **Scalability:** Master-slave replication and clustering for growth.
- **Data Structures:** Supports diverse types for varied uses.
- **Persistence:** RDB snapshots and AOF logs ensure durability.
- **Integration:** Seamless integration reduces tuning efforts.

What should users look for in Redis reviews?

- **Performance:** Real-time data retrieval boosts efficiency.
- **Cost-Effectiveness:** Stable, budget-friendly implementation.
- **Versatility:** Suitable for diverse environments and platforms.
- **Reliability:** Ensures data persistence and high availability.

Redis finds application across industries for tasks like caching to improve application performance and speed, minimizing database load. It enables real-time processing for session storage, push notifications, and analytics. As a messaging platform, Redis handles high traffic and supports replication and clustering for cross-platform scalability.

Valuable Features

Excerpts from real customer reviews on PeerSpot:



“Redis is a powerful and reliable tool for improving application performance.”



RituRaj

SDE 2 at Virtusa



“Overall, Redis helps us design more efficient applications by improving performance while allowing our backend services to scale more effectively.”



Verified user

Senior Software Engineer at a consultancy with 1-10 employees



“Since I started using Redis, I feel that the product is saving me some performance tuning time.”



KarimGarchi

Database Admin and Architect at D-EDGE Hospitality Solutions



“Redis is good for distributed caching management.”



Yaseer Arafat

.Net Developer at Freelance



“Redis has multiple valuable features such as being a free and reliable open-source tool.”



Verified user

Data Engineer at a photography company with 1,001-5,000 employees



“It is particularly efficient for cloud-based storage and operations.”



Vasu Bansal

Student at Otto-von-Guericke-Universitaet



“Redis provides an easy setup and operation process, allowing users to quickly connect and use it without hassle.”



Rotem Fogel

R&D Director at a tech vendor with 201-500 employees

What users had to say about valuable features:

Redis has multiple valuable features such as being a free and reliable open-source tool. It functions similarly to a foundational building block in a larger system, enabling native integration and high functionality in core data processes. Despite its limitations, Redis provides valuable performance enhancement through system fine-tuning and multi-thread handling.

Verified user

[Read full review](#) 

Data Engineer at a photography company with 1,001-5,000 employees

“I find Redis valuable primarily for its caching capabilities, particularly in handling cache requests effectively. Its simplicity in managing key-value pairs for caching is one of its strengths, making it a preferred choice over more complex databases like MongoDB for specific use cases. However, I haven't explored Redis extensively for managing complex data structures beyond caching, as MongoDB might be more suitable for such scenarios.

.”

Simon Lazarev

[Read full review](#) 

Founder at Proton Technologies

Redis stands out for its extremely fast in-memory performance, support for rich data structures such as string, hash, and list, and features such as TTL for automatic expiration. It is also very useful for caching, sessions management, and rate limiting. I rely mostly on the fast memory performance combined with caching, which helps reduce database load and improve response time for frequently accessed data.

Redis has played a key role in improving system scalability and performance. By offloading frequent reads from the database and enabling fast in-memory cache access, it reduced latency, improved throughput, and helped maintain stability during peak loads. .”

RituRaj

SDE 2 at Virtusa

[Read full review](#) 

“Caching is the best feature that Redis offers. The most valuable features include high-speed in-memory data access, flexible data structures, caching capabilities, data expiration and time-to-live management, high availability and scalability, and atomic operations. These are the biggest use cases that I find in a low-latency tool like Redis.

“The features of high-speed in-memory data access and flexible data structures are very impactful in my work. We have been using them throughout to help cache frequently accessed data, which reduces the number of database calls and improves API response times, especially for high-traffic scenarios. Redis has also simplified scaling because multiple application instances can share the same cached data instead of each instance maintaining its own state. This makes deployments more reliable and helps the team focus on business features rather than performance optimization issues. Overall, Redis has helped us build faster, more scalable applications..”

Verified user

Senior Software Engineer at a consultancy with 1-10 employees

[Read full review](#) 

“The best features of Redis, from my personal perspective, are the performance, which is very quick, and it's very simple to implement.

“Since I started using Redis, I feel that the product is saving me some performance tuning time. It's very easy, I have few parameters to tune, and it seems to have performance without a lot of working on the performance, compared to Cassandra, where you have to configure the memory and many other settings.

“The integration capability of Redis is excellent.

“Redis is very affordable because it's free..”

KarimGarchi

Database Admin and Architect at D-EDGE Hospitality Solutions

[Read full review](#) 

“Redis has many good features that I find useful. The general caching use case is the most important, along with high performance, rich data structures, persistence options, easy integration, and good scalability. What I appreciate most is its speed and ease of use. In practice, Redis dramatically reduces database load by serving frequently accessed data from memory, and features like automatic key expiration, known as TTL, along with persistence options, make it reliable for production applications while remaining simple to work with.

Speed makes the biggest difference in reducing response times in my projects. In one project, the application displays a user dashboard that requires several database queries to gather profile information, preferences, and recent activity. Instead of executing those queries on every request, I cache the assembled dashboard data in Redis. As a result, repeat requests are served directly from memory, making the dashboard load almost instantly while significantly reducing the load on the database. Regarding the TTL feature, it is especially useful because it keeps cached data fresh automatically. For example, I cache product listings and search results with a TTL of around five to ten minutes. If a user requests the same data within that period, Redis returns it immediately. Once the TTL expires, Redis automatically removes the cache entry, and the next request regenerates the data from the database. This gives me a good balance between performance and data freshness without needing custom cache cleanup logic, which would need to be done manually and might feel much more tedious and difficult. Redis handles it very effectively.

Using Redis has positively impacted my organization, especially regarding performance because most data is fetched very quickly without having to execute numerous database queries in the first place. Redis has had a positive impact by improving application performance, reducing database load, and making the system more scalable. By caching frequently accessed data, I am able to serve requests much faster, which improves the overall user experience. After introducing Redis for caching dashboard and search data, users experience noticeably faster page loads on repeat requests. The database handles fewer read operations, which improves overall system responsiveness and provides more headroom during peak usage. Redis has helped me avoid unnecessary repeated

database queries for pages that users have visited previously..”

Verified user

Product Engineer at a tech vendor with 11-50 employees

[Read full review](#) 

Other Solutions Considered

“In my projects, we use documents basically, so all the NoSQL databases can be mapped with an API to have a kind of independence from Redis and any tool. If tomorrow we want to move from Redis to something better, we are independent from that..”

KarimGarchi

Database Admin and Architect at D-EDGE Hospitality Solutions

[Read full review](#) 

“I considered a couple of options before choosing Redis. I looked at a few alternative approaches depending on the use case, mainly in-memory caching. For in-memory caching inside the application, I used Caffeine or Guava as local caches. I initially considered local in-memory caching in the application. It was not sufficient due to cache inconsistency, data loss on restart, and difficulty in horizontal scaling. This is the main reason why I shifted to Redis..”

Verified user

Product Engineer at a tech vendor with 11-50 employees

[Read full review](#) 

Before choosing Redis, we mainly relied on database-level caching or direct queries. As the application scaled, it started impacting performance, so we switched to Redis for its speed and better caching capabilities.

Before Redis, we relied on the normal database, but before we considered Redis, we looked at a few alternatives such as Memcached. Redis stood out because of its richer data structures and additional features such as persistence and pub/sub features. .”

RituRaj

SDE 2 at Virtusa

[Read full review](#) 

“Before using Redis, we relied more on application-level caching and database approaches for managing frequently accessed data. As our applications and workloads grew, those approaches became less efficient and harder to scale. The main reason we moved to Redis is that it provides a dedicated, high-performance caching layer with low-latency access, built-in expiration management, and support for distributed scenarios. It simplified our architecture by providing a reliable solution for caching, temporary data storage, and shared application status across multiple instances. The main reason for switching was improved performance, reduced database load, better scalability, and easier management of cached data..”

Verified user

Senior Software Engineer at a consultancy with 1-10 employees

[Read full review](#) 

“Before Redis, my team tested a couple of different solutions. The main driver for switching to Redis was scalability and performance. Local caching helped somewhat, but it did not work effectively in distributed systems. Redis gave me a centralized, fast, and reliable caching layer that significantly improved system performance and consistency. Initially, my team typically relied on direct database queries and sometimes in-memory caching at the application level, using simple hash maps or local caches such as Caffeine or Guava in Java applications. The higher database load during peak traffic is why I shifted to Redis in the first place..”

Verified user

[Read full review](#) 

Product Engineer at a tech vendor with 11-50 employees

“Our team evaluated multiple options before choosing Redis, including Memcached, normal database caching, and cloud-native caching solutions such as AWS ElastiCache. The main factors we considered were performance, scalability, reliability, ease of integration, supported data structure, and operational simplicity. Redis stood out because it provides more than just simple caching features, such as multiple data structures, time-to-live management, atomic operations, persistence options, and support for distributed workloads, which made it a more flexible solution for our application needs. Overall, Redis provided the right balance of performance, flexibility, and ecosystem support compared to the other options we evaluated, such as Memcached and AWS ElastiCache..”

Verified user

[Read full review](#) 

Senior Software Engineer at a consultancy with 1-10 employees

ROI

Real user quotes about their ROI:

We have seen a strong ROI after implementing Redis. We reduced the database read load by around 30 to 40 percent and improved API response time by 20 to 30 percent, specifically for frequently accessed endpoints.

RituRaj

SDE 2 at Virtusa

[Read full review](#) 

“The solution does what I need, and I have no issues with it. Since I didn't install Redis and don't need to maintain it, the service saves me a lot of time..”

David Shlingbaum

IT Development Manager, Architect, Developer at Miltel Communications LTD

[Read full review](#) 

“Eventually, we need to manage caching. If we manage on our own, it will take a lot of developer resources, infrastructure, and environmental resources. So, it is better to use Redis. The ROI is better..”

IdoBen Ami

Software Architect at Yum! Brands

[Read full review](#) 

“By introducing Redis as a caching layer for frequently accessed data, we reduce repeated database queries and improve response time for high-read APIs. This allowed our system to handle higher traffic more efficiently without requiring equivalent increases in database capacity. We have seen a positive return on investment from using Redis, mainly through improved application performance, reduced database load, and lower operational overhead. Although we have not measured a direct reduction in headcount or specific dollar saving, the productivity and scalability benefits have been valuable..”

Verified user

Senior Software Engineer at a consultancy with 1-10 employees

[Read full review](#) 

“I have not seen a direct ROI concerning fewer employees needed because caching does not mean that I can reduce the number of employees. I did not track a direct monetary ROI in exact figures, but I observed a clear operational and performance-based return that indirectly translated into cost savings and efficiency gains. By caching frequently accessed data in Redis, I significantly reduced the number of database queries, resulting in savings concerning infrastructure costs. This is an indirect ROI. Redis improved system scalability, allowing me to handle higher traffic without proportional infrastructure growth, which creates scalability ROI. Although I did not measure ROI in exact dollar terms, the combined effect of reduced database load, improved performance, and lower operational overhead clearly translated into cost efficiency and better scalability. If I had to scale without Redis, infrastructure costs and engineering effort would have been significantly higher..”

Verified user

Product Engineer at a tech vendor with 11-50 employees

[Read full review](#) 

“Measuring the return on investment (ROI) of Redis can vary based on the specific use case, but here are some general observations:

1. **Improved Performance:** The speed and efficiency of Redis can lead to significant performance improvements. This results in faster response times and a better user experience, which can translate into higher user retention and satisfaction.
2. **Cost Savings:** By reducing the load on your primary databases and improving application performance, Redis can help lower infrastructure costs. This is especially true if you're using a managed service that scales with demand.
3. **Developer Productivity:** The simplicity and versatility of Redis can increase developer productivity. Faster implementation of features and reduced time spent on performance optimization can lead to quicker time-to-market for new products or features.
4. **Scalability:** Redis's ability to scale efficiently ensures that your application can handle increased traffic without significant additional investment. This scalability helps maintain performance and reliability as your user base grows.
5. **Reliability:** The stability and high availability of Redis can reduce downtime and data loss, leading to more consistent service and higher customer trust.

While the specific ROI will depend on your unique circumstances, these benefits often result in significant value for organizations that implement Redis effectively. .”

Yaseer Arafat

.Net Developer at Freelance

[Read full review](#) 

Use Case

We use Redis for several purposes, including ranking, counting, saving, sharing, caching, and setting time-to-live notifications. These functionalities are employed across various AI projects and in data processing tools, where Redis helps with the ongoing data pipeline process.

Verified user

Data Engineer at a photography company with 1,001-5,000 employees

[Read full review](#) 

“Redis is used for a part of a booking engine for travel, specifically for the front part to get some sessions and information about the sessions. If a customer or user is using the sites in different parts, we use Redis to get this information in cache..”

KarimGarchi

Database Admin and Architect at D-EDGE Hospitality Solutions

[Read full review](#) 

“I'm the only user utilizing Redis in my company, as my platform is relatively new and has around fifty users. However, in my previous role at Profit Clicks, we had approximately ten thousand users. We utilized a combination of databases, including Redis, MongoDB, and MariaDB. We needed to handle push notifications for our users, sending them three times a day, and managing a large volume of notifications efficiently was crucial. Redis played a role in managing this workload.

.”

Simon Lazarev

Founder at Proton Technologies

[Read full review](#) 

My main use case for Redis is caching to improve application performance and reduce database load.

One specific example from my backend services is using Redis to cache frequently accessed data like product details. Instead of querying the database every time, the application first checks Redis. If data is present, it returns instantly, which significantly reduces the database load and improves response time.

“Apart from cache, I have also used Redis for session storage and rate limiting. It helps in managing user sessions efficiently and controlling traffic spikes, which improves overall system reliability. .”

RituRaj

SDE 2 at Virtusa

[Read full review](#) 

“The main use case for Redis is caching. Caching is the primary purpose of using Redis for my organization, as I use it to cache frequently accessed data to improve application performance and reduce database load, functioning as a caching service for database optimization. For example, when a user requests their personalized dashboard, the application first checks Redis for the cached dashboard. If it is available, the response is returned immediately. If not, the application retrieves the data from the database, stores it in Redis with a short expiration time of five to ten minutes, and then returns it to the user. This significantly reduces database queries and improves response times.

I also use Redis for session management and short-lived application state. For example, during an AI chat session, Redis stores conversation context, rate-limiting counters, or temporary workflow state, which can be accessed quickly without repeatedly querying the primary database..”

Verified user

Product Engineer at a tech vendor with 11-50 employees

[Read full review](#) 

“Redis is used primarily as a caching layer to provide a high-performance caching solution that improves application response times and reduces load on backend services and databases. We use it mainly to store frequently accessed data and temporary session information and short-lived application state where low-latency access is important. It also helps with scenarios such as rate limiting, distributed caching across multiple application instances, and improving scalability for APIs that experience frequent reads.

“One example of how I use Redis in my applications is using it as a caching layer for frequently accessed application data instead of querying the database every time an API request comes in. We cache commonly requested data in Redis with an appropriate expiration period. For example, in one of our APIs, frequently accessed user or configuration data is stored in Redis after the first request, so subsequent requests can retrieve the data directly from Redis, which significantly reduces database load and improves API response times, especially during periods of high traffic. Redis has also been useful for distributed scenarios where multiple application instances need to share temporary state, ensuring consistent behavior when workloads scale horizontally.

“Beyond caching, Redis has become useful as a supporting component for improving application scalability and reliability. It helps us reduce unnecessary database operations, improve response times, and provide a consistent, low-latency data access layer for frequently used information. Another benefit is its flexibility for handling temporary data, such as session-related information, rate limiting, and short-lived application state, where using a traditional database would add unnecessary overhead. Overall, Redis helps us design more efficient applications by improving performance while allowing our backend services to scale more effectively..”

Verified user[Read full review](#) 

Senior Software Engineer at a consultancy with 1-10 employees

Setup

The setup process involves configuring and preparing the product or service for use, which may include tasks such as installation, account creation, initial configuration, and troubleshooting any issues that may arise. Below you can find real user quotes about the setup process.

The initial setup of Redis was difficult, with a rating of two or three out of ten. A deep understanding of Redis's core and high technical knowledge was required, making the process lengthy and complex.

Verified user

[Read full review](#) 

Data Engineer at a photography company with 1,001-5,000 employees

“Redis requires occasional maintenance, but it is generally a low-maintenance solution and is straightforward to manage. I haven't encountered any difficulties or downtime after updates, as Redis is robust and concise..”

Simon Lazarev

[Read full review](#) 

Founder at Proton Technologies

“Using AWS for Redis, the setup involves going through a user-friendly UI, making the process simple and straightforward. After setup, you get an endpoint, and it's ready to use..”

Baudilio Garcia

Head of Platforms at Shalion

[Read full review](#) 

“Our experience with Redis's pricing, setup cost, and licensing has been positive overall. The setup process was straightforward, and integrating Redis into our existing application architecture did not require significant additional infrastructure changes. The licensing options provide flexibility depending on the deployment model and scale requirements. The main value comes from the performance improvements, reduced database load, and increased scalability that Redis provides. From a cost perspective, the investment is justified by the operational benefits, including improved application responsiveness and reduced efforts spent managing performance issues..”

Verified user

Senior Software Engineer at a consultancy with 1-10 employees

[Read full review](#) 

“Regarding my experience with pricing, setup cost, and licensing for Redis, pricing and setup cost are handled by a particular team, but I have some knowledge. Redis is open source at its core, so there is no traditional licensing cost for the basic Redis engine itself. There is no licensing cost because the open-source version is free to use. When it comes to setup cost, it is relatively low, especially when using managed services. Since I self-host it, costs come from infrastructure. Additional efforts for high availability setup, monitoring, backup configuration, and higher DevOps overhead would be management costs. Pricing mainly depends on memory size, throughput, and availability configuration..”

Verified user

[Read full review](#) 

Product Engineer at a tech vendor with 11-50 employees

“The initial setup of Redis was relatively straightforward, especially for basic configurations. Here are a few points that highlight the process:

1. **Installation:** Installing Redis was simple, with clear documentation and guides available for various operating systems. Whether deploying on a local machine or a cloud instance, the steps were easy to follow.
2. **Configuration:** For basic use cases, the default configuration settings were sufficient. However, for more complex setups involving clustering or high availability, the configuration required more attention and understanding of Redis parameters.
3. **Integration:** Integrating Redis with our existing applications was smooth, thanks to the availability of client libraries for different programming languages. This made it easier to incorporate Redis into our tech stack.
4. **Learning Curve:** While the basic commands and operations were easy to grasp, understanding advanced features and optimizing performance required more in-depth knowledge. Fortunately, Redis's comprehensive documentation and community support helped bridge this gap.

Overall, the initial setup was manageable and straightforward for basic use, with some complexity arising in advanced configurations..”

Yaseer Arafat

.Net Developer at Freelance

[Read full review](#) 

Customer Service and Support

The customer support I have experienced has been good overall. Since Redis is quite stable and well-documented, we have not needed much support, but when required, the response has been helpful.

RituRaj

SDE 2 at Virtusa

[Read full review](#) 

“We did not encounter anything with their customer support because we have not fallen into situations where we had to contact them. By simply referring to their documentation, we have been able to fix our bugs and general issues. In that instance, their documentation is so good that we do not really have to reach out to their customer support..”

Verified user

Senior Software Engineer at a consultancy with 1-10 employees

[Read full review](#) 

“I have not had the chance to interact with customer support because there have not been any problems requiring it. I mostly rely on Redis documentation and community support rather than direct vendor support, especially when using the open-source version or managed cloud services, because its documentation is well-documented..”

Verified user

Product Engineer at a tech vendor with 11-50 employees

[Read full review](#) 

Other Advice

I rate Redis seven out of ten overall. While it's a powerful open-source tool, it has areas needing improvement in terms of scalability and certain functionalities. Despite this, the tool provides reliability for our needs. I recommend considering these aspects before adopting Redis for large-scale operations, especially if high technical competencies are needed.

Verified user

Data Engineer at a photography company with 1,001-5,000 employees

[Read full review](#) 

“I haven't explored AI-driven initiatives yet, but they are on my road map. I would recommend Redis for all kinds of caching and message brokering needs. It's a reliable and straightforward solution.

Overall, I would rate Redis ten out of ten..”

Simon Lazarev

Founder at Proton Technologies

[Read full review](#) 

“If Redis has questions or comments related to my review, it's possible for them to reach me via email to clarify something.

“I am interested in being a reference for Redis.

“On a scale of 1–10, I rate Redis a 10..”

KarimGarchi

Database Admin and Architect at D-EDGE Hospitality Solutions

[Read full review](#) 

“If somebody is looking to use Redis, it is much easier and more scalable for making most database queries faster, especially for repeated pages while fetching data. It helps tremendously in handling multiple queries and fetching in a cluster system or in complex systems. My suggestion is that if somebody is looking for a platform to cache or render repeated queries efficiently, Redis is the proper tool.

Redis is a great tool. It is a good caching tool and a supporting layer, not a tool that I can use as a direct database. Although I could, there are better options for that. Redis is made for a particular reason, specifically for caching purposes. If I am using it for that particular use case, then I will benefit the most. I give this review a rating of eight out of ten..”

Verified user

Product Engineer at a tech vendor with 11-50 employees

[Read full review](#) 

Redis is very reliable and easy to integrate. Its simplicity combined with the performance makes it a great choice for backend developers.

My advice would be to first clearly define your use cases, specifically for caching or

real-time scenarios, and also pay attention to memory management. Choose the right eviction policies and implement proper monitoring from the beginning. Plan for memory optimization, set appropriate TTLs, and implement strong monitoring and alerting for stability at any scale.

“Redis is a powerful and reliable tool for improving application performance. Its speed and flexibility make it a great choice for modern backend systems. It significantly improves performance and scalability with proper planning. It works very effectively for high-traffic applications. I would rate this product an 8 out of 10. .”

RituRaj

SDE 2 at Virtusa

[Read full review](#) 

“There are a couple of things to consider when using Redis. It is a supporting layer, not a main database. Identifying specific use cases where Redis can provide the most value, such as caching, session management, rate limiting, real-time data processing, or reducing database load, is crucial. Redis is very powerful, but it should be used with a clear caching and data management strategy rather than as a replacement for every data store. For production workloads, I recommend setting up proper monitoring, security controls, high availability, and capacity planning from the beginning.

“Redis is a great caching tool, and anyone can try it or at least see whether it fits their organization. It fits our organization, so it might definitely fit for somebody else as well with the features that have been provided. I suggest trying it out to see whether it is the right solution for your needs. If it fits you, then you are good to go. I would rate this product nine out of ten..”

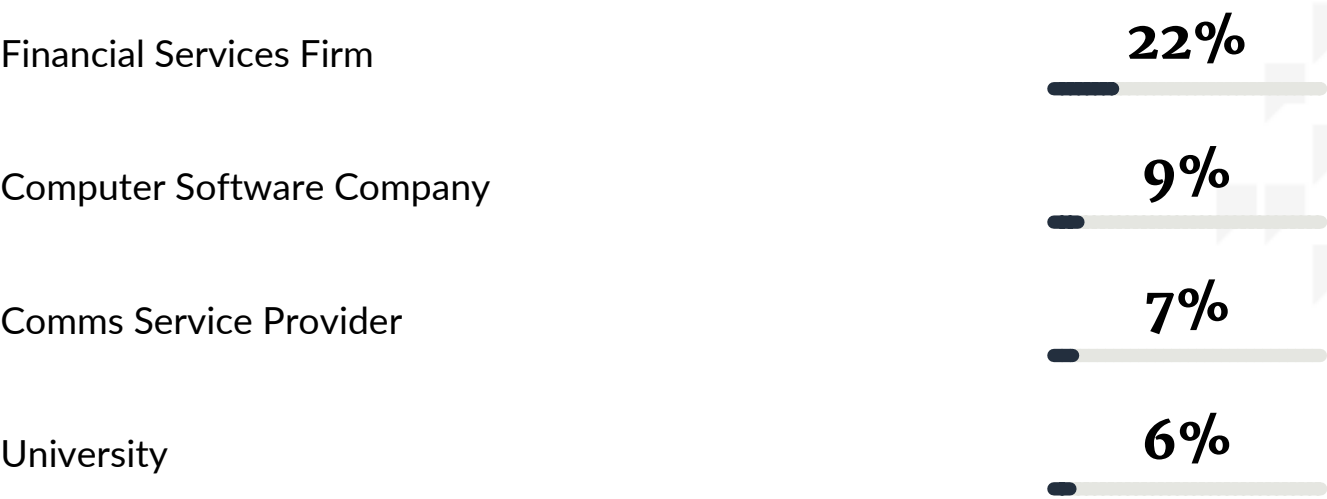
Verified user

Senior Software Engineer at a consultancy with 1-10 employees

[Read full review](#) 

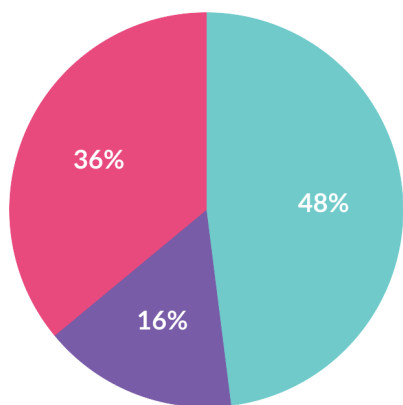
Top Industries

by visitors reading reviews

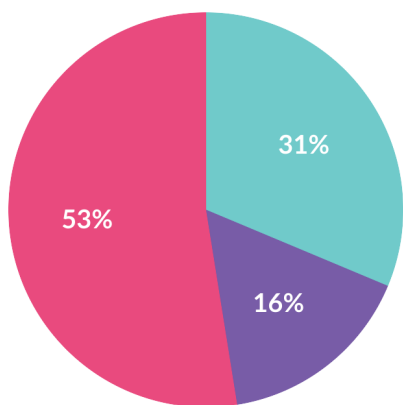


Company Size

by reviewers



by visitors reading reviews



 Large Enterprise  Midsize Enterprise  Small Business

About this buyer's guide

Thanks for downloading this PeerSpot report.

The summaries, overviews and recaps in this report are all based on real user feedback and reviews collected by PeerSpot's team. Every reviewer on PeerSpot has been authenticated with our triple authentication process. This is done to ensure that every review provided is an unbiased review from a real user.

Get a custom version of this report... Personalized for you!

Please note that this is a generic report based on reviews and opinions from the collective PeerSpot community. We offer a [customized report](#) of solutions recommended for you based on:

- Your industry
- Company size
- Which solutions you're already considering

The customized report will include recommendations for you based on what other people like you are using and researching.

Answer a few questions in our short wizard to get your customized report.

[Get your personalized report here](#)

About PeerSpot

PeerSpot is the leading review site for cloud, AI, and business software. We created PeerSpot to provide a trusted platform to share information about software, applications, and services. Since 2012, over 22 million people have used PeerSpot to choose the right software for their business.

PeerSpot helps tech professionals by providing:

- A list of products recommended by real users
- In-depth reviews, including pros and cons
- Specific information to help you choose the best vendor for your needs

Use PeerSpot to:

- Read and post reviews of products
- Access over 30,000 buyer's guides and comparison reports
- Request or share information about functionality, quality, and pricing

Join PeerSpot to connect with peers to help you:

- Get immediate answers to questions
- Validate vendor claims
- Exchange tips for getting the best deals with vendor

Visit PeerSpot: www.peerspot.com

PeerSpot

244 5th Avenue, Suite R-230 • New York, NY 10001

reports@peerspot.com

+1 646.328.1944