

aws marketplace

Hasura

Reviews, tips, and advice from real users



Powered by  PeerSpot



Contents

- Product Recap..... 3 - 4
- Valuable Features..... 5 - 12
- Other Solutions Considered..... 13 - 14
- ROI..... 15 - 16
- Use Case..... 17 - 20
- Setup..... 21 - 22
- Other Advice..... 23 - 25
- Trends..... 26 - 27
- About PeerSpot..... 28 - 29

Product Recap



Hasura

Hasura Recap

Hasura is a GraphQL engine that automates the creation of APIs, providing developers with a real-time data source integration. It streamlines backend development and accelerates the time-to-market for applications by enabling instant GraphQL API generation from existing databases.

Hasura empowers developers with its ability to automatically generate GraphQL APIs from databases, offering a solution for quickly developing applications that require real-time capabilities. It allows seamless integration with existing databases, providing scalability and comprehensive access control that caters to enterprise-level demands. This transforms database queries into efficient GraphQL operations, simplifying data management and reducing backend coding efforts significantly.

What are Hasura's most important features?

- **Real-Time GraphQL:** Provides real-time capabilities with GraphQL subscriptions.
- **Data Federation:** Enables integration across multiple data sources seamlessly.
- **Permissions & Access Control:** Offers granular security measures for data access.
- **Remote Schemas:** Supports schema stitching to extend GraphQL APIs easily.

What benefits or ROI can users expect from Hasura?

- **Reduced Development Time:** Speeds up application development with auto-generated APIs.
- **Scalability:** Adapts easily to growing data needs and user demands.
- **Cost Efficiency:** Decreases overhead by minimizing the need for custom backend coding.
- **Increased Flexibility:** Supports diverse architectures through robust API management.

In industries like e-commerce and fintech, Hasura's real-time data capabilities have revolutionized user experiences. By offering instant data updates and seamless database integration, companies can enhance customer interactions and data analysis. Access control features support compliance with industry-specific regulations, ensuring secure data handling across various sectors.

Valuable Features

Excerpts from real customer reviews on PeerSpot:

- ✓ “Hasura impacts our organization positively by being extremely useful.”



Yogi V

Senior Software Engineer at Pine Labs

- ✓ “The most significant benefit I have seen with Hasura is that it makes the developer's job much easier by eliminating the need to write custom APIs and schemas.”



Tejesh Reddy

Software Engineer at Sunlighten

- ✓ “Hasura offers excellent features including role-based filtering, which eliminates the N+1 problem, making it easy to maintain schemas and ensuring fast retrieval and fast mutations.”



Amith V

Senior Software Engineer at Pit Solutions

- ✔ “Hasura has positively impacted my organization because for this project, whatever project I'm working on where previously we were using the normal DB2 which was a little complicated to use, Hasura made it very easy to understand the data flowing and data integration.”



Vijayalakshmi Ankalagi

Business Analyst at a healthcare company with 201-500 employees

- ✔ “Hasura's best feature is that it provides an API gateway, and it also provides remote schemas and monitoring, which are the best methods they are offering.”



Shivam Chauhan

Manager, Software Engineering at a tech vendor with 10,001+ employees

- ✔ “My advice to others looking into using Hasura is to directly use it since it is a very good application with high performance and scalability.”



Fatha Mohamed Qureshi

Lead Software Engineer at Siemens

- ✔ “Earlier, we took around maybe two days to create some APIs and set up DB and other things; in Hasura we can create tables and relationships and APIs in the end, and DB creation, configuration, and deletion of rows are very easy, so where two days were taken, now we can do it in half to one day at max, and in development effort, very much time is saved and the execution is fast.”



Durrez Ahmed

Software engineer at a consultancy with 201-500 employees

What users had to say about valuable features:

“The best features Hasura offers include the ability to directly integrate it with the front end, and the performance is quite good and fast.

“Hasura has positively impacted my organization by boosting the productivity of the application in real time. It has also worked with real-time subscriptions where the performance was good, and queries were very fast..”

Fatha Mohamed Qureshi

Lead Software Engineer at Siemens

[Read full review](#)

“The best features Hasura offers include a simple UI that anybody can understand, and that is very helpful.

I appreciate the layout of the UI and the way queries are created; the product development of Hasura is very easy to understand.

Hasura has positively impacted my organization, as the majority of the APIs were integrated through GraphQL, and most of the time we are using Hasura.

It saves the team time; I'm not sure about the cost. Since I'm a front-end developer with limited knowledge in the back end, I can easily get the queries directly from the database, making it a win-win for both the team and time..”

Verified user

Senior Software Engineer at a tech vendor with 1,001-5,000 employees

[Read full review](#) 

“In my opinion, the best features Hasura offers are the auto-generated GraphQL APIs, as it has auto-generated GraphQL APIs and instantly provides queries, mutations, and real-time subscriptions. The frontend can fetch whatever data it needs, instead of fetching huge payloads. Since we used AG Grid for showing huge datasets in a table, the filtering and querying capabilities were very powerful, and the role-based security implementation with Hasura is also commendable.

“Hasura positively impacted my organization by facilitating fast development; we saved a lot of time in development. I can estimate that we saved significant time; we do not need to manually create all the GraphQL queries or API generations. I can say we could build an API within hours instead of taking days.

“The time saved reflects a significant return on investment..”

Verified user

Full Stack Developer at a tech services company with 11-50 employees

[Read full review](#) 

“Hasura offers quite user-friendly features. It is easy to adapt, especially since it is GraphQL, which is similar to SQL. It provides a nice interface to build different APIs, and one feature I really appreciated was access control. For example, we had many customers with one single table of data, but data access control prevented one customer from seeing another customer's data.

“Access control was vital for our product-based company with more than 100 customers to prevent data breaches and leaks, which are security issues tied to customer trust. To maintain everything and the trust of our customers, it was crucial to have good data access control. Via this API access control, it was easy; we just added company-level access control to Hasura, filtering based on different customers, ensuring each customer received only their data.

“Hasura positively impacted our organization through useful access control, allowing us to directly obtain data without extra layers of access control processes, which increased efficiency and speed..”

Deepak Swain

Senior Data Engineer at a energy/utilities company with 1,001-5,000 employees

[Read full review](#) 

“The features I appreciate most about Hasura include the ability to connect to any database; we are using PostgreSQL, and it automatically generates schemas once connected to the tables. Hasura reads the tables and creates the schemas itself, eliminating the need for users to write their own schemas. Furthermore, it allows for authentication in multiple ways, such as Webhooks and JWT tokens, providing role-based authentication for data, which is a significant feature for us.

“We do use Hasura's real-time subscriptions, and they have a substantial impact on our application's interactivity because, with subscriptions, there is no need to refresh the app whenever data changes. Hasura's subscription feature automatically updates data, making it very important for our workflow. All features provided by GraphQL are provided by Hasura, including subscriptions, mutations, and queries.

“Hasura's granular data access control helps in securing sensitive information by implementing role-based authentication. For instance, if a user is set up with their username or user ID and email, they only have access to their data and nothing else, unless they are an admin or receive other specific permissions..”

Tejesh Reddy

Software Engineer at Sunlighten

[Read full review](#) 

“There is an issue with HTTP transactions since HTTP is a stateless protocol, making it impossible to maintain transactions, which is the only drawback found in the GraphQL queries. Custom transaction methods were built to address this, and role-based query filters were also used, adding connections with the subgraph and Apollo Federated graph, treating it as a subgraph.

Hasura offers excellent features including role-based filtering, which eliminates the N+1 problem, making it easy to maintain schemas and ensuring fast retrieval and fast mutations.


The biggest feature that made the most difference for our team is schema maintenance. Hasura is really good for handling different migrations and schema changes. A custom migration task was built for Azure DevOps, which makes it easy to manage. Hasura engine also keeps track of migration numbers, making it simple to handle migrations, and adjusting role-based visibility for users when exposing certain fields or columns is straightforward.

Hasura has positively impacted the organization mainly by enabling fast retrieval of data through GraphQL, which is the main reason for using it.

Faster development cycles were noticed when comparing direct SQL queries and the overhead in the execution pipeline. Moving to the GraphQL query made data retrieval very easy..”

Amith V

Senior Software Engineer at Pit Solutions

[Read full review](#) 

Other Solutions Considered

“Other options like Hot Chocolate and ChilliCream were evaluated since the application is being built on top of .NET Core, but Hasura seemed easier to maintain..”

Amith V

Senior Software Engineer at Pit Solutions

[Read full review](#) 

“I did not evaluate other options before choosing Hasura because when I joined this company, they had already introduced Hasura. I am not sure if they did any research on alternatives, but I can confirm that I did not..”

Tejesh Reddy

Software Engineer at Sunlighten

[Read full review](#) 

“Before choosing Hasura, I evaluated PostgreSQL DB, which we connected directly from the code. This required us to write the code, so we opted for Hasura so the productivity would be faster..”

Fatha Mohamed Qureshi

Lead Software Engineer at Siemens

[Read full review](#) 

“I did not previously use a different solution with this company.

“Before choosing Hasura, we evaluated other options; we traditionally used REST APIs by building backend services in ASP.NET Web API, along with using Node.js, Express, and Java Spring Boot. The old approach created too much boilerplate, requiring a controller, service, repository, and mapping validations for every table and making our frontend heavily dependent on the backend..”

Verified user

Full Stack Developer at a tech services company with 11-50 employees

[Read full review](#) 

ROI

Real user quotes about their ROI:

I have not seen a return on investment with Hasura because as a team member I used that one, and I'm not the best person to answer that. My manager would be better positioned to provide that information.

Vijayalakshmi Ankalagi

Business Analyst at a healthcare company with 201-500 employees

[Read full review](#) 

“I have seen a return on investment with Hasura, as the time saved is significant at 70% less time dealing with database queries and REST APIs for development, which saved a lot of people's time..”

Verified user


DevOps Engineer at a tech services company with 51-200 employees

[Read full review](#) 

“I have seen a return on investment. When we write code, the code runs for some milliseconds, and then we have the database query that runs for some milliseconds as well. Using Hasura avoids writing any extra code, so we could save those milliseconds on the server, which will save cost. We can save millions of dollars, or thousands of dollars at least..”

Fatha Mohamed Qureshi

Lead Software Engineer at Siemens

[Read full review](#) 

Use Case

“My main use case for Hasura is database connectivity and creating APIs. I can use an API for the database quickly without writing any code, which is a specific example of how I use Hasura for database connectivity and API creation..”

Fatha Mohamed Qureshi

Lead Software Engineer at Siemens

[Read full review](#) 

“As an end-user of Hasura, we are using it within our projects. Our usual use cases for Hasura involve using it as an authorizer; instead of writing our own GraphQL schemas, we use Hasura, which is 10x faster, and we don't need to write as much logic ourselves. Hasura is a GraphQL engine tool that generates schemas automatically to connect with the database, so we do not need to manually write the schemas and APIs. We can directly integrate through our application, making the developer's job easier. Additionally, it connects very fast, with APIs being 10x faster compared to normal APIs..”

Tejesh Reddy

Software Engineer at Sunlighten

[Read full review](#) 

“I use Hasura for development purposes as a GraphQL engine that provides REST endpoints for data retrieval and mutation, integrated with our .NET application. It is incredibly fast for development because it combines GraphQL queries directly into a single SQL statement.

Hasura was used for the article pipeline page where GraphQL queries were created for fetching large scale data using the GraphQL endpoint directly from Hasura engine. With a million records of data, the retrieval is completed within a second using the GraphQL query..”

Amith V

Senior Software Engineer at Pit Solutions

[Read full review](#) 

“Hasura was my main tool and platform for exposing all of our APIs for our insights page. Everything on our insights webpage was our analytics products from Atency, and that is where we were using Hasura for data access control, having security classes, and controlling the data via APIs.

“In general, everything related to APIs was via Hasura in my previous company. We had our webpage with reports via Power BI, but for all our API-related components, it was everything via Hasura..”

Deepak Swain

Senior Data Engineer at a energy/utilities company with 1,001-5,000 employees

[Read full review](#) 

“My main use case for Hasura is to write GraphQL queries that I use in the implementation of APIs on the front end.

I implemented a feature called OKR, Objectives and Key Results, and I used Hasura to write all the GraphQL queries, which have been very helpful for me. Without any back-end developer's help, I can create my own APIs with a limited version of key and data values.

As a front-end engineer with limited knowledge in the back end, since I started using Hasura, I have no dependency on back-end developers, so I don't need to wait for them to create APIs. I can create my own GraphQL queries, which are very simple, or mutations, and I use subscriptions for real-time data retrieval, which is very helpful for me..”

Verified user

Senior Software Engineer at a tech vendor with 1,001-5,000 employees

[Read full review](#) 

“My main use case for Hasura involves dealing with multiple source systems, where we had to create a bunch of repetitive backend work, leading to many APIs that needed to be built. Hasura made development easier because it was useful without wasting time building basic APIs over and over again; instead of creating the controller, service, repository, and DTOs manually, along with filtering and sorting, Hasura has made the work straightforward.


“A quick specific example of a project where Hasura helped me save time is our auto-renewal system, which we built in a recent renewal job. Without Hasura, after processing, a backend would need to expose an API endpoint such as /renewals, requiring us to manually build the DTOs, maintain multiple endpoints, support frontend changes, and handle real-time updates separately using WebSockets. With Hasura, the work became straightforward; the .NET jobs update the database, and it automatically exposes the updated data with no need for backend changes, also aiding in the real-time dashboard updates.

“The feature that had the most impact on my work was real-time data combined with eliminating the backend API development; that combination greatly sped up building the systems. This mattered most because our system needs to process tasks such as renewal processing, background jobs, and orchestration workflows. Using the AG Grid table was particularly useful, as we have to show renewals that are pending, processing, completed, or failed. Without Hasura, we would have had to create several REST endpoints, polling APIs, and manage WebSockets, along with manual filtering APIs. Hasura has changed our approach by handling API generation, GraphQL queries, subscriptions, and filtering, allowing the frontend to no longer wait for backend APIs to make changes.

“Hasura reduced backend boilerplate and enabled real-time operational dashboards by automatically exposing database changes through GraphQL subscriptions, allowing our services to focus only on business processing while Hasura handled API generation and live data..”

Verified user

Full Stack Developer at a tech services company with 11-50 employees

[Read full review](#) 

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 Hasura was very easy. Once redirected to the dashboard, I can add a new project by clicking a button for a new project. It asks if I want to go with a free plan or a professional plan, and if I select the free plan, I provide a provider such as Google Cloud, Azure, or AWS. Based on this, I can create a new project and add my DB connection string, such as CockroachDB. Just put it in and click save to create the connection, after which I can launch the console..”

Shivam Chauhan

Manager, Software Engineering at a tech vendor with 10,001+ employees

[Read full review](#) 

“Hasura's integration capabilities with other tools or services I use are easier to integrate. It is a pretty easy setup; once we have a subscription, we can plug in the database credentials and then get started with creating the GraphQL. This is a much easier process compared to manually creating those endpoints at a backend.

“Hasura saves a lot of time for my team; just for a single table implementation, it would have taken one or two days, but now it just comes down to one or two minutes.

“Hasura's monitoring and logging capabilities are sufficient for my needs; we can integrate Prometheus and Grafana for monitoring. I collected all the logs and metrics from Prometheus and connected them to the Grafana dashboard, from where I can visualize all the issues that are occurring..”

Verified user

[Read full review](#) 

DevOps Engineer at a tech services company with 51-200 employees

Other Advice

I would give the advice that if others are looking to build something fast and want to avoid the overhead of writing boilerplate APIs and implementing RBAC, they can definitely go for Hasura because it saves a lot of time. I give this product a review rating of seven out of ten.

Moitreyo Chakraborty

Software Developer at a tech vendor with 10,001+ employees

[Read full review](#) 

“My advice to others looking into using Hasura is that if they have an automated system such as auto-renewal, or multiple source systems to deal with while using GraphQL, then I would suggest going with Hasura. I rate this product a 9..”

Verified user

Full Stack Developer at a tech services company with 11-50 employees

[Read full review](#) 

“I suggest to beginners looking into using Hasura who want to learn the back end, or for simple applications that want to build APIs, that instead of going for a REST API, I suggest GraphQL through Hasura. That will be easy to understand, easy to implement, and it's very beginner-friendly for small applications. I gave this product a rating of eight out of ten..”

Verified user

Senior Software Engineer at a tech vendor with 1,001-5,000 employees

[Read full review](#) 

“My experience while working with it was quite satisfactory.

“For others considering Hasura, its effectiveness varies depending on the company size and data amount. For medium-scale or startup companies, it is an excellent option since it is free, but it may not handle immense data for huge companies. I would rate Hasura as a seven out of ten based on my experience because it has some limitations, and I would have preferred more features, such as building complex queries, which are not possible in Hasura..”

Deepak Swain

Senior Data Engineer at a energy/utilities company with 1,001-5,000 employees

[Read full review](#) 

“If I had to pick just one number, I would say nine because development would be faster, which is expected in any industry. It requires less time for development.

“Regarding Hasura's AI capabilities, I think its governance and security are good. Perhaps we could increase those in the future, but since it is in the initial phase, it is quite good with no issues as of now.

“The accuracy of Hasura's output is quite good as of now, with no issues with that.

“My advice to others looking into using Hasura is to directly use it since it is a very good application with high performance and scalability.

“Since Hasura is quite a good product, I have also suggested using it in different products in the current organization. I hope for the best. I would rate this review a 9 overall..”

Fatha Mohamed Qureshi

Lead Software Engineer at Siemens

[Read full review](#) 

“To evaluate the effectiveness of Hasura's event triggers, I use metrics such as the time taken for API calls, which we check using Postman, measuring in milliseconds.

“I can only assess Hasura's value in integrating with microservices architectures from our experience, as we are only connecting one microservice with one [PostgreSQL](#) database. I cannot speak to how it performs with multiple microservices; however, it integrates very effectively with the database, and the process is straightforward.

“The most significant benefit I have seen with Hasura is that it makes the developer's job much easier by eliminating the need to write custom APIs and schemas. It is 10x faster because they maintain some internal cache mechanisms efficiently.

“I have used Hasura's official documentation, which is well-provided. If you are a skilled engineer, you can comprehend it effectively, but you need to know the jargon. I would give the documentation a rating of 7 because if someone is not that skilled, they may struggle to understand it without prior knowledge.

“I would rate this review an overall 8..”

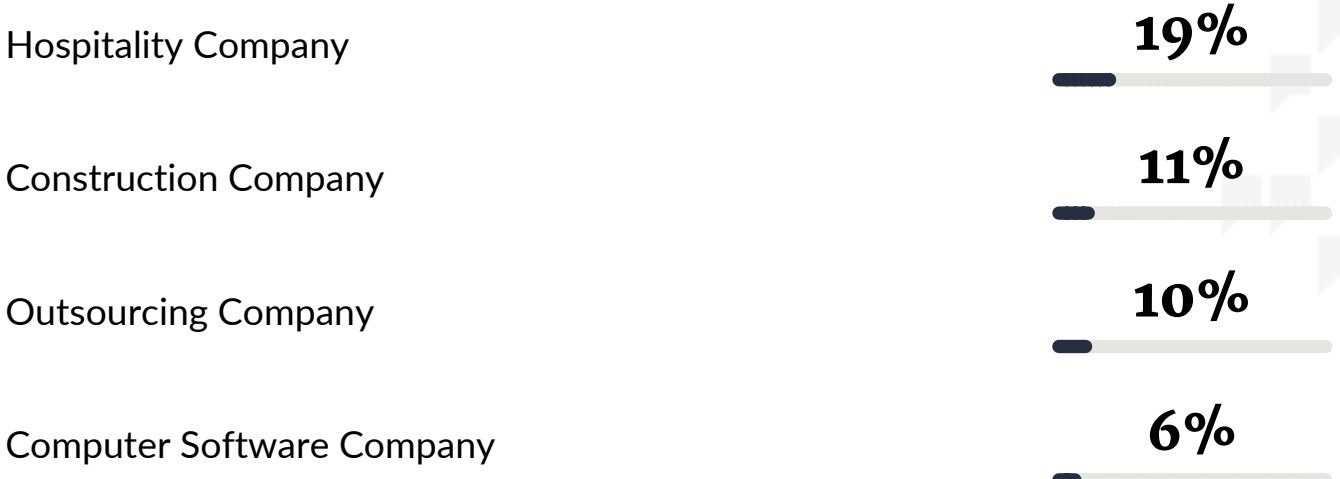
Tejesh Reddy

©2026 PeerSpot. All Rights Reserved

[Read full review](#)  25

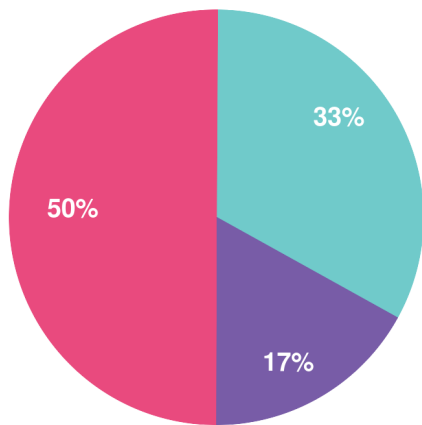
Top Industries

by visitors reading reviews

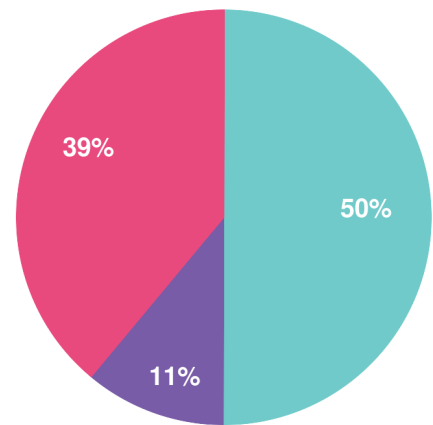


Company Size

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