International Journal of Multidisciplinary Engineering in Current Research - IJMEC
Volume 8, Issue 12, December-2023, http://ijmec.com/, ISSN: 2456-4265

REVIEW OF DATA MANAGEMENT INFRASTRUCTURE IN CLOUDSCENARIO

Mareboina Sriram, Ms.V. Prasanthi

[1]B.tech Student, Department Of Electronics and Computer Engineering, J.B Institute of Engineering and Technology

[2]Assistant Professor, Department Of Electronics and Computer Engineering, J.B Institute of Engineering and Technology

Abstract: As a cloud computing systems refers to the demand delivery of IT resources via the internetwith pay as you go pricing. A cloud offers many services to the end users such as software,infrastructure and platform go on. To develop the scalable techniques for transaction management utilizing the snapshot isolation (SI) model. The model can lead to non- serializable transaction executions investigate two conflict detection techniques for ensuring serializability. To support scalability and investigate system architectures and mechanisms in which the transaction management functions are decoupled from the storagesystem and integrated with the application-level processes. We present system architecturesand demonstrate their scalability under the scale-out model of cloud computing platforms.In the first system architecture all transaction management functions are executed in a fully decentralized manner by the application processes. To perform a comparative evaluation ofthese architectures using the benchmark and demonstrate their scalability.

I. INTRODUCTION

The entire sequence of steps is not performed as single atomic actions are not performing all steps of the commit protocol as one atomic action raises a number of issues. The transaction management protocol should ensure the transactional consistency when multiple processes execute the protocol steps concurrently. Any of these steps may get interrupted due to process crashes or delayed due to slow execution. To address this problem, the transaction managementprotocol should support a model of cooperative recovery. Any process should be able to complete any partially executed sequence of commit/abort actions on behalf of another processthat is suspected to be failed.

In the intricate landscape of distributed systems, the execution of transaction commit protocols is often far from a seamless, atomic affair. Attempting to perform all steps of the commit protocol as a single atomic action presents a host of challenges and potential pitfalls. The inherent complexity is exacerbated when multiple processes concurrently engage in executing these protocol steps. The asynchronous nature of these transactions introduces the risk of interruptions caused by process crashes or delays arising from slow execution. Recognizing these challenges, an indispensable component of a robust transaction management protocol is the incorporation of a cooperative recovery model. This model is designed to mitigate the uncertainties introduced by interruptions, crashes, or delays. In essence, it empowers any process within the distributed system to step in and seamlessly complete any partially executed sequence of commit or abort actions on behalf of another process suspected to have failed. By enabling such cooperative recovery mechanisms, the protocol not only bolsters the fault tolerance of the system but also ensures transactional consistency in the faceof concurrent and potentially disrupted protocol executions. This approach underscores the resilience needed in the dynamic and unpredictable environment of distributed computing, offering a pragmatic solution to the challenges inherent in transaction management across multiple processes.

The execution of a transaction commit protocol involves a series of distinct steps, and it is impractical to treat them as a single atomic action due to various challenges. Performing all commit protocol steps atomically raises concerns regarding the potential for interruptions caused by process crashes or delays in execution. In order to ensure transactional consistency when multiple processes concurrently execute these protocol steps, a robust transaction management protocol is essential. This protocol must be designed to accommodate the possibility of interruptions, providing a mechanism for cooperative recovery. In this context, any process should have the capability to finalize any partially executed commit/abort sequence on behalf of another process suspected to have failed. By incorporatingthis model of cooperative recovery, the transaction management protocol enhances the resilience of the system, mitigating the impact of process failures and contributing to the overall reliability of transactional operations in a distributed computing environment

II. LITERATURE SURVEY

A comprehensive literature survey on the analysis of data storage and transaction in the cloud environment reveals a dynamic and evolving landscape driven by the rapid advancements in cloud computing technologies. Researchers have extensively explored various aspects of cloud-based data storage, emphasizing scalability, reliability, and security. Numerous studies focus on the challenges posed by the increasing volume of data and the need for efficient storage solutions. Transaction analysis within the cloud is another critical area, with scholars investigating transactional consistency, reliability, and performance in distributed environments. Security concerns, including data encryption and access control, have been recurrent themes in the literature, reflecting the growing importance of safeguarding sensitive information in the cloud. Furthermore, the integration of emerging technologies such as blockchain for secure transactions and distributed ledger systems has gained attention. As cloudcomputing continues to play a pivotal role in modern IT infrastructures, the literature underscores the ongoing efforts to enhance data storage and transactional processes to meet theevolving needs of users while addressing the associated challenges

Moreover, transaction analysis in the cloud is intricately tied to ensuringrobustness and consistency in distributed systems. Studies often investigate the nuances of transactional models, exploring concepts like ACID properties (Atomicity, Consistency, Isolation, Durability) and their applicability in cloud environments. The advent of serverless computing and microservices architectures has also spurred research into transactional consistency models that can accommodate the decentralized nature of modern cloud applications.

Security, being a paramount concern, extends beyond traditional measures to include in-depth explorations of encryption mechanisms, identity management, and secure data sharing protocols. Cryptographic techniques, such as homomorphic encryption, are scrutinized for their efficacy in preserving data privacy in the cloud. Additionally, access control mechanisms are continuously refined to adapt to the dynamic nature of cloud environments, with a particular emphasis on fine-grained access policies.

As cloud technologies continue to advance, researchers are exploring the potential of artificial intelligence and machine learning for data storage optimization, predictive maintenance, and anomaly detection in transactions. The evolving nature of these technologies prompts a continuous reevaluation of existing models and the development of innovative strategies to ensure the seamless functioning of cloud-based storageand transactions in an increasingly complex and interconnected digital landscape.

Proposed System

The proposed system for the analysis of data storage and transactions in the cloud environment envisions a comprehensive and innovative approach to address the existing challenges and enhance the overall efficiency and security of cloud-based operations. One key aspect of the proposed system is an advanced security framework that integrates state-of-the-art encryption algorithms, robust access control mechanisms, and continuous monitoringprotocols. This aims to fortify data integrity and confidentiality, mitigating the risks associated with unauthorized access and potential breaches. The system also incorporates a dynamic and scalable architecture, leveraging cutting-edge technologies such as distributed ledger systems or blockchain to enhance transactional consistency and transparency. By embracing a decentralized approach, the proposed system aims to alleviate issues related to latency and network congestion, fostering improved performance in data storage and transaction analysis.

Furthermore, the proposed system places a strong emphasis on interoperability and compatibility, seeking to establish standardized protocols for seamless integration with various cloud platforms and services. This approach not only facilitates smooth data exchange but also ensures adaptability in multi-cloud or hybrid cloud environments. Additionally, the system incorporates machine learning algorithms to optimizeresource utilization, predict potential system failures, and automate decision-making processes for more efficient and cost-effective cloud operations.

In terms of compliance and regulatory adherence, the proposed system includes features that enable organizations to navigate diverse legal requirements seamlessly. This includes tools for automated compliance checks, audit trails, and configurable policies toalign with evolving regulatory landscapes. The user interface is designed with user- friendliness in mind, providing intuitive controls for administrators to manage and monitor data storage and transactions effectively.

Overall, the proposed system envisions a holistic and forward-looking solution that not only mitigates the disadvantages of existing systems but also embraces emerging technologies to meet the evolving needs of cloud-based data storage and transaction analysis. Through its integrated approach to security, scalability, interoperability, and compliance, the proposed system strives to set a new standard for robust, efficient, and securecloud operations.

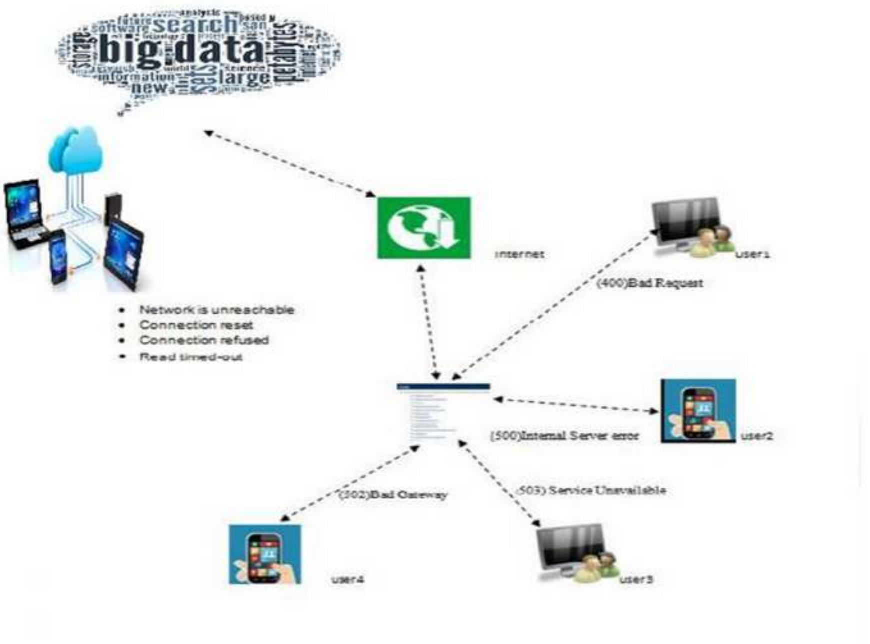


Fig 2.4.1 : Proposed SYSTEM ARCHITECTURE

III. ANALYSIS

The Review of data management infrastructure in cloud scenario is a critical examination of the methodologies, technologies, and challenges associated with managing and processing data in cloud-based systems. This multifaceted analysis involves an in-depth exploration of the existing practices, focusing on aspects such as scalability, reliability, security, and efficiency. Researchers and industry experts scrutinize the strengths and weaknesses of current systems, delving into topics ranging from traditional relational databases to contemporary NoSQL solutions. Transactional analysis becomes a pivotal point of evaluation, encompassing considerations of consistency, isolation, and durability in distributed cloud environments. Security, a paramount concern in cloud computing, involves a comprehensive review of encryption mechanisms, access controls, and compliance measures. The analysis also extends to emerging technologies like blockchain and their integration into data storage and transactional processes for enhanced transparency and security. Scalability challenges, interoperability issues, and the impact of evolving regulatory landscapes are scrutinized to provide a comprehensive understanding of the complexities surrounding cloud-based data management. Through this analysis, researchers aim to contribute insights and recommendations that can guide the development of more resilient, secure, and efficient systems for data storage and transactions in the ever-evolving cloud environment.

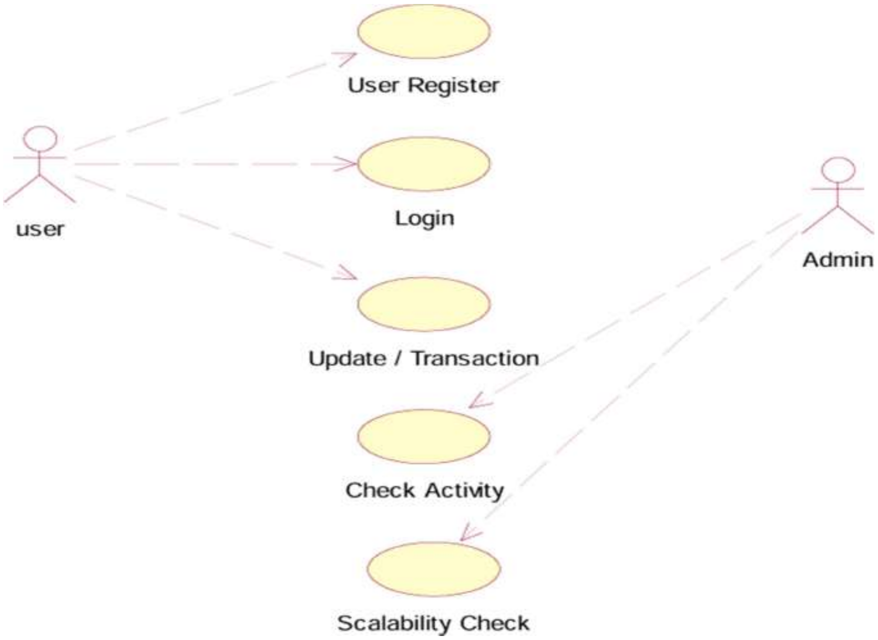
IV. DESIGN

The design for the analysis of data storage and transactions in the cloud environment involvesa comprehensive and strategic approach to address the complexities of cloud-based systems. The architectural framework must prioritize scalability, accommodating the dynamic nature of data volumes in the cloud. A robust security architecture is fundamental, encompassing encryption protocols, secure access controls, and monitoring mechanisms to safeguard sensitive information. Transactional analysis design should focus on ensuring consistency andreliability in distributed environments, potentially incorporating blockchain or distributed ledger technologies for enhanced transparency and security. The system's design should facilitate seamless interoperability with various cloud platforms and services, acknowledging the diverse landscape of cloud technologies. Integration of machine learning algorithms becomes crucial for optimizing resource utilization, predicting system behavior, and automating decision-making processes. Compliance considerations should be embedded within the design, ensuring adherence to legal and regulatory frameworks governing data storage and transactions. An intuitive user interface design should empower administrators with efficient tools for monitoring, managing, and extracting meaningful insights from the analyzed data. Overall, a well-conceived design for data storage and transaction analysis in thecloud should be adaptable, secure, and performance-oriented, catering to the evolving demandsof modern cloud computing environments.

DFD OR UML DIAGRAMS

USE CASE DIAGRAM:

A use case diagram is a type of behavioral diagram created from a Use-case analysis. The purpose of use case is to present overview of the functionality provided by the system in terms of actors, their goals and any dependencies between those use cases.



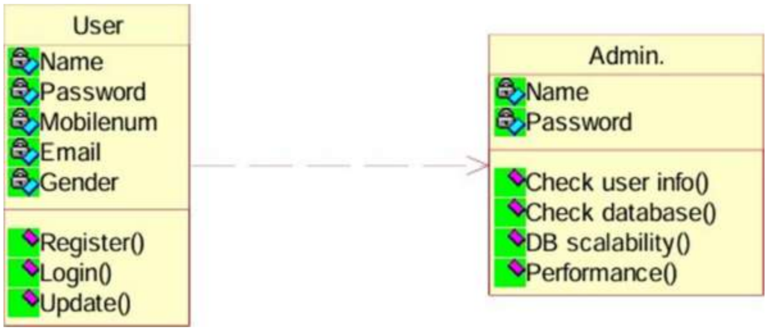
4.1.1 use case diagram

CLASS DIAGRAM:

A class diagram in the UML is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, and the relationships between the classes.

Private visibility hides information from anything outside the class partition.

Public visibility allows all other classes to view the marked information.



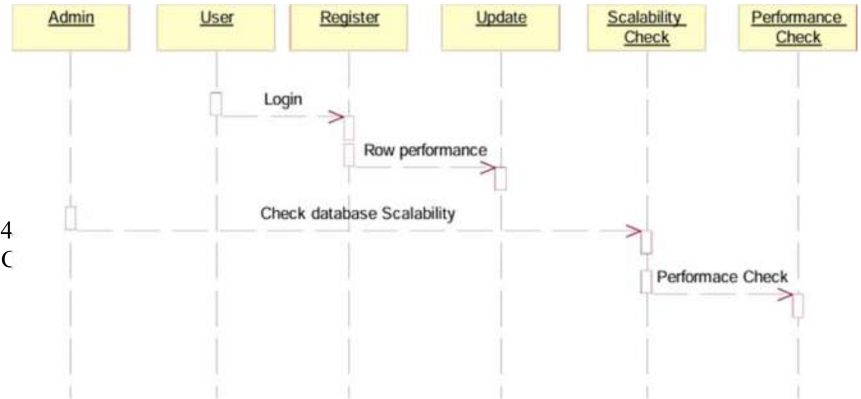
class diagram

SEQUENCE DIAGRAM:

A sequence diagram in UML is a kind of interaction diagram that shows how the processesoperate with one another and in what order.

It is a construct of a message sequence chart. Sequence diagrams aresometimes called Event-trace diagrams, event scenarios, and timing diagrams.

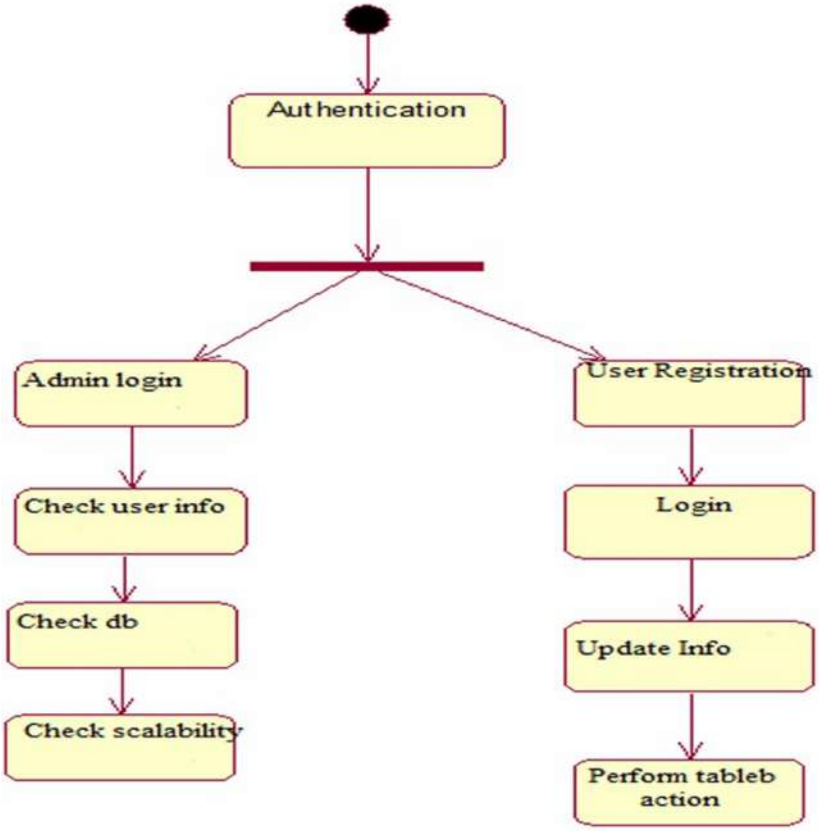
The below diagram shows the sequence flow shows how theprocess occurs in this project.



sequence diagram

ACTIVITY DIAGRAM:

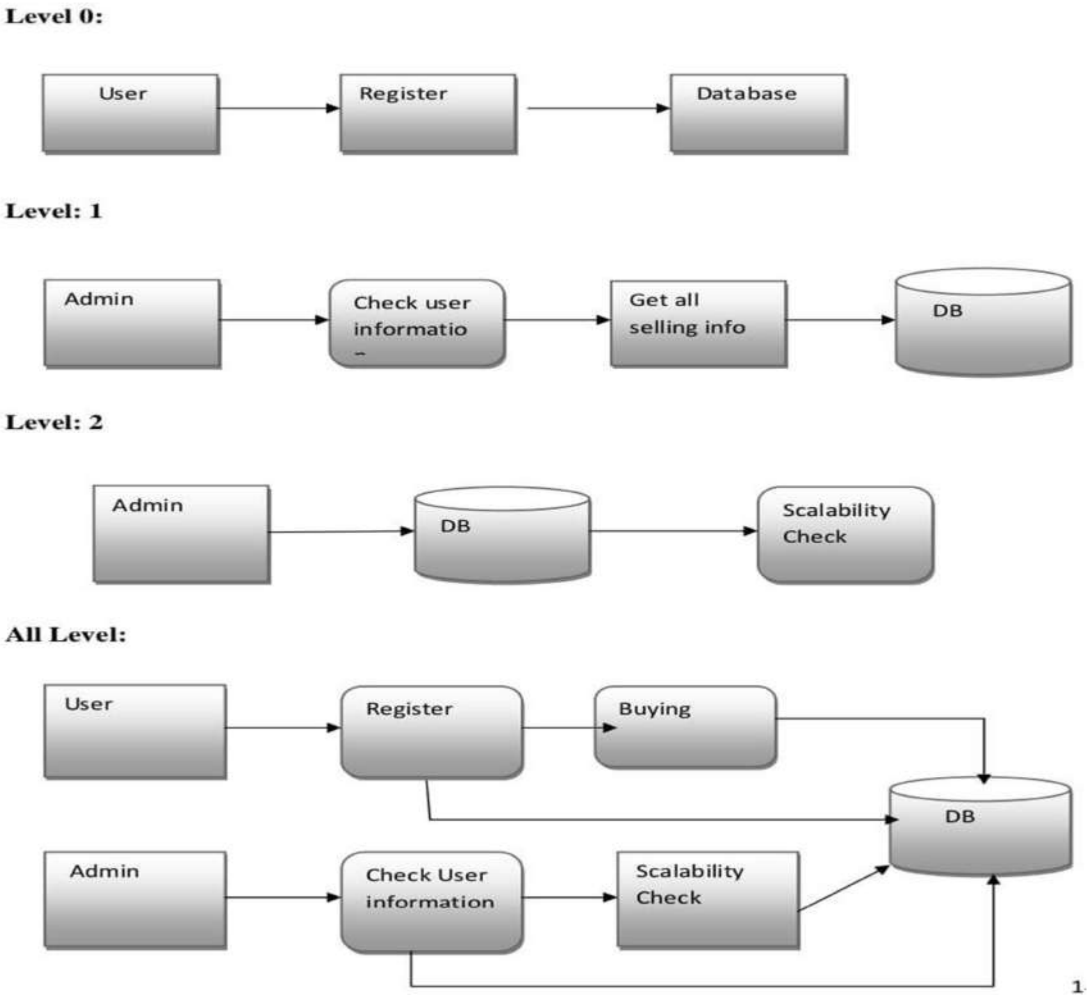
Activity diagram are a loosely defined diagram to show workflows of stepwise activities and actions, with support for choice, iteration and concurrency. UML, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. UML activity diagrams could potentially model the internal logic of a complex operation. In many ways UML activity diagrams are the object-oriented equivalent of flow charts and data flow diagrams (DFDs) from structural development.



ACTIVITY DIAGRAM

DATA FLOW DIAGRAM:

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. It differs from the flowchart as it shows the data flow instead of the control flow of the program. A data flow diagram can also be used for the visualization of data processing. The DFD is designed to show how a system is divided into smaller portions and tohighlight the flow of data between those parts.



V. IMPLEMENTATION AND RESULTS

The implementation and results phase of the analysis of data storage and transactions in the cloud environment marks the culmination of meticulous planning and development efforts, transitioning theoretical concepts into tangible, operational solutions. This pivotal stage involves deploying the designed system into the cloud environment, putting the data storage and transaction analysis components into action. The implementation process requires carefulconsideration of scalability, security, and interoperability to ensure a seamless integration withthe cloud infrastructure. Once deployed, the system undergoes rigorous testing to validate its functionality, security measures, and performance under diverse scenarios. Following successful implementation, the focus shifts to analyzing the results generated by the system. This involves assessing the efficiency of data storage mechanisms, transactional consistency, and the overall performance of the cloud-based solution. The results obtained provide valuable insights into the system's strengths and areas for improvement, guiding further refinements orenhancements. This phase is crucial in gauging the practical effectiveness of the data storage and transactional analysis in the dynamic context of the cloud, ultimately contributing to informed decision-making and potential optimizations for future iterations.

OUTPUT SCREENS

User Login Page:

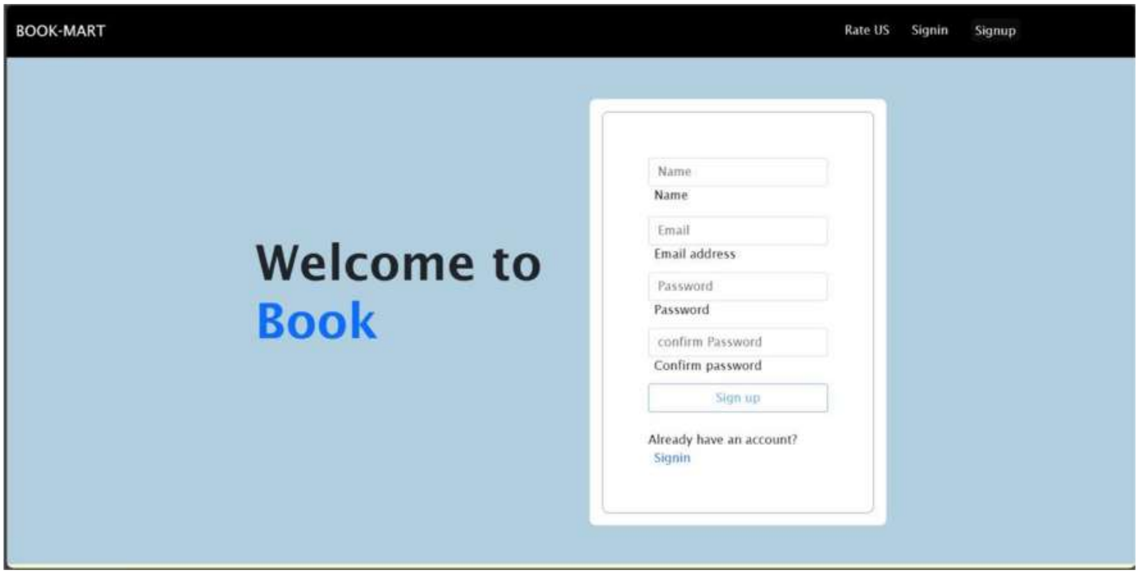
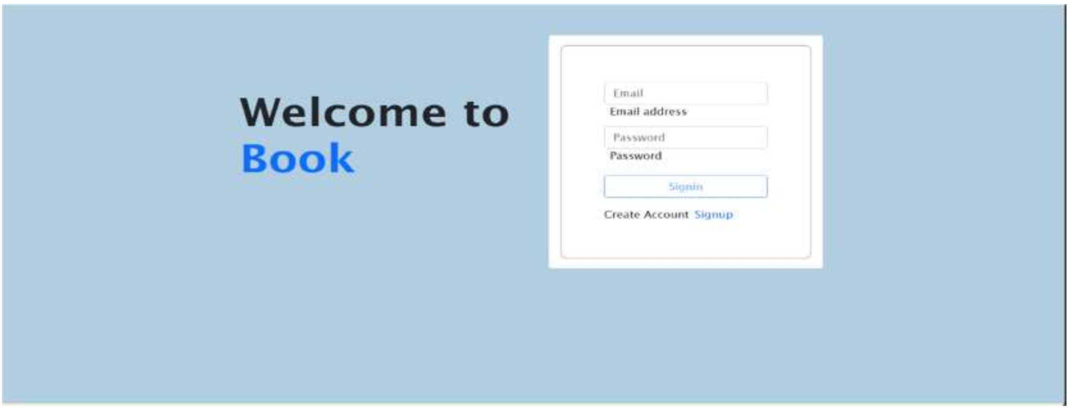


Fig 5.4.1

Admin Login Page:



User Interface:

Fig 5.4.2

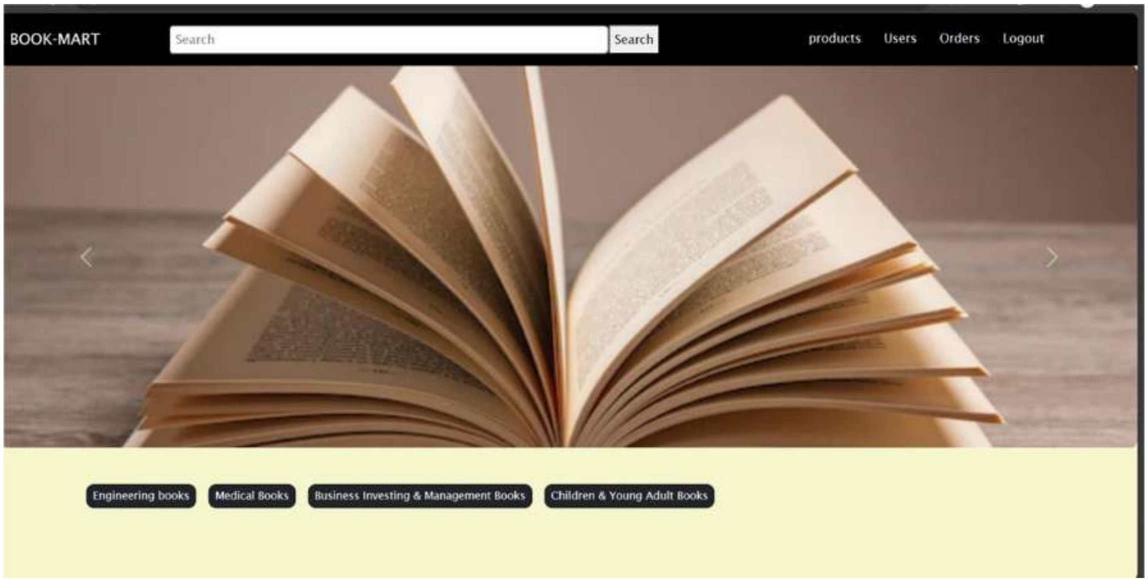


Fig 5.4.3

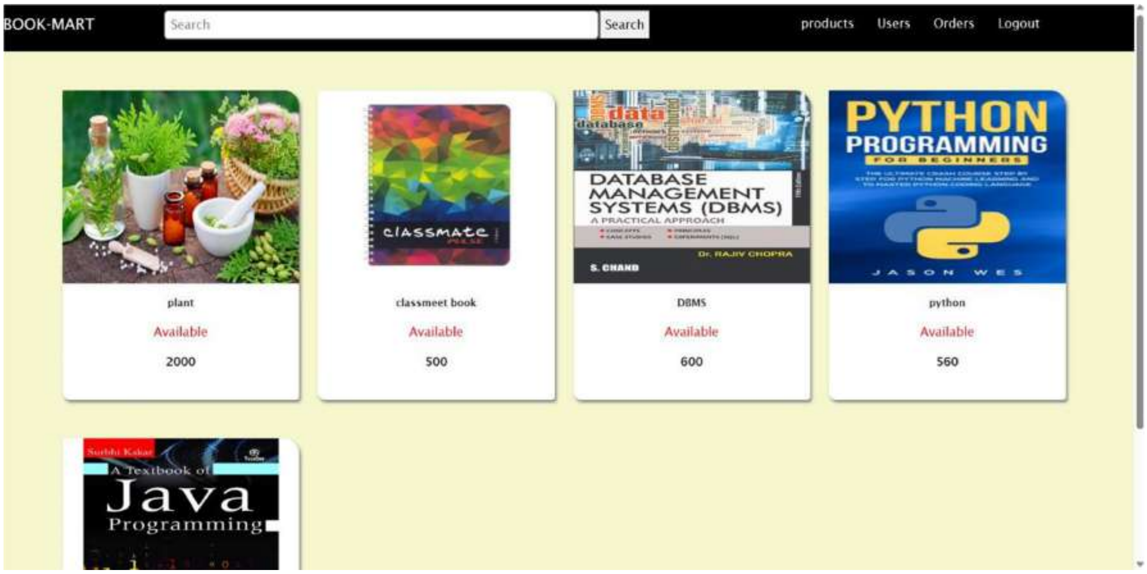
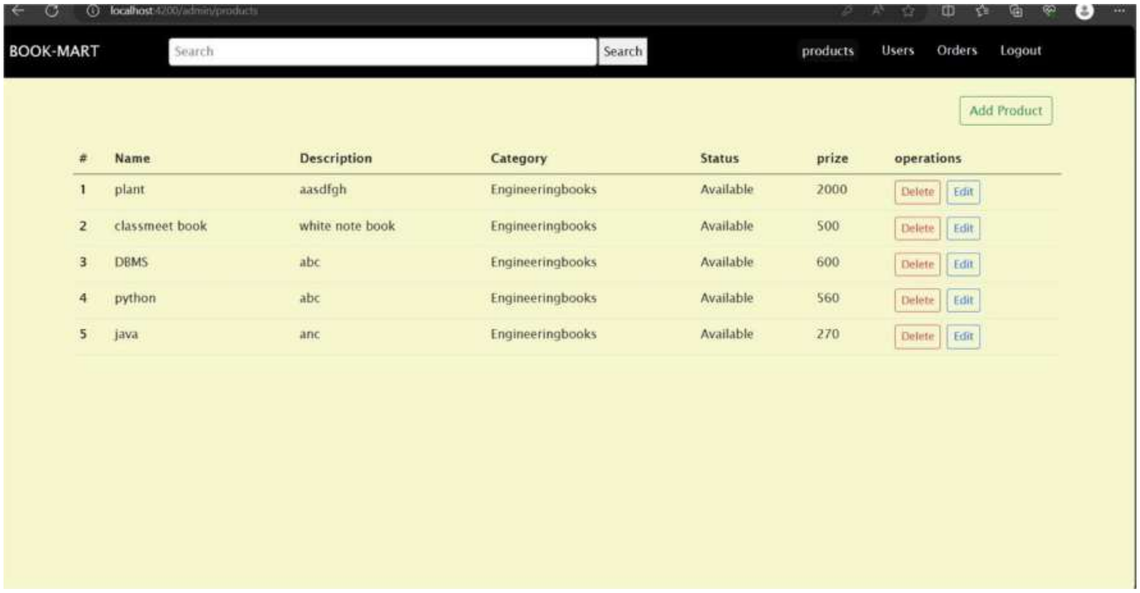


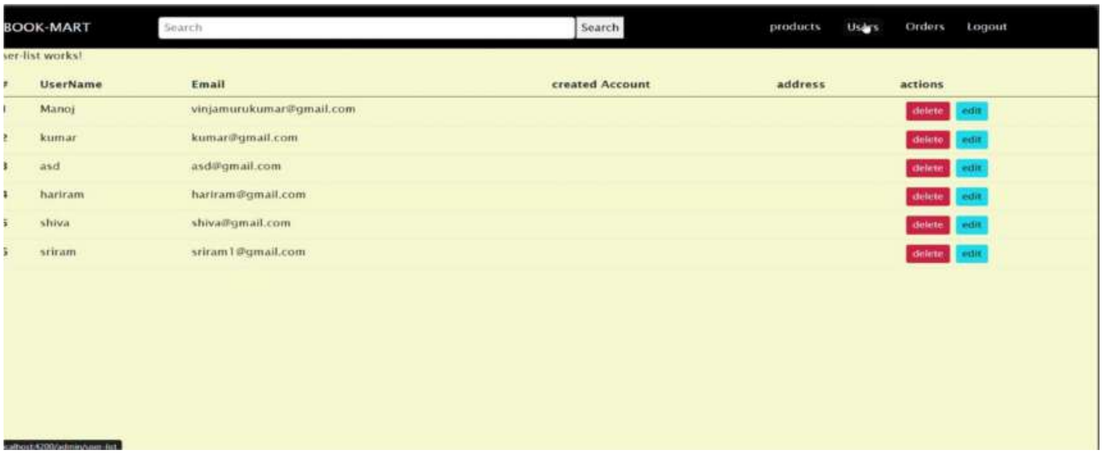
Fig 5.4.4

Admin User Interface:



#	Name	Description	Category	Status	prize	operations
1	plant	aasdfgh	Engineeringbooks	Available	2000	Delete Edit
2	classmeet book	white note book	Engineeringbooks	Available	500	Delete Edit
3	DBMS	abc	Engineeringbooks	Available	600	Delete Edit
4	python	abc	Engineeringbooks	Available	560	Delete Edit
5	java	anc	Engineeringbooks	Available	270	Delete Edit

Fig 5.4.5

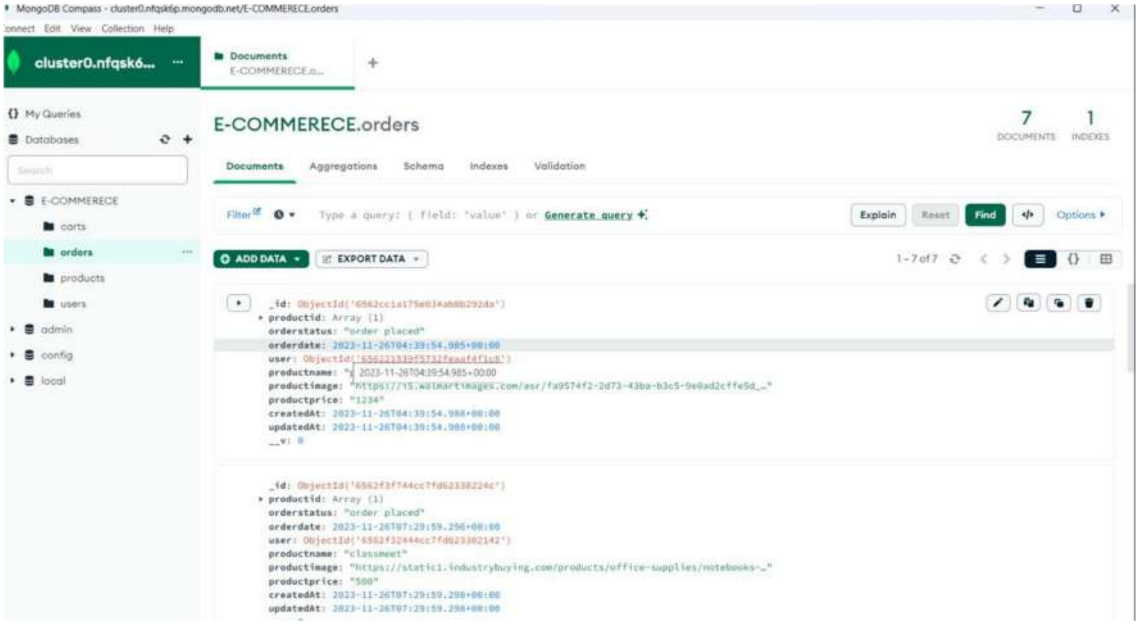


ser-list works!

#	UserName	Email	created Account	address	actions
1	Manoj	vinjamurukumar@gmail.com			delete edit
2	kumar	kumar@gmail.com			delete edit
3	asd	asd@gmail.com			delete edit
4	hariram	hariram@gmail.com			delete edit
5	shiva	shiva@gmail.com			delete edit
6	sriram	sriram1@gmail.com			delete edit

Fig 5.4.6

Data stored in mongoDBc:Order Data:

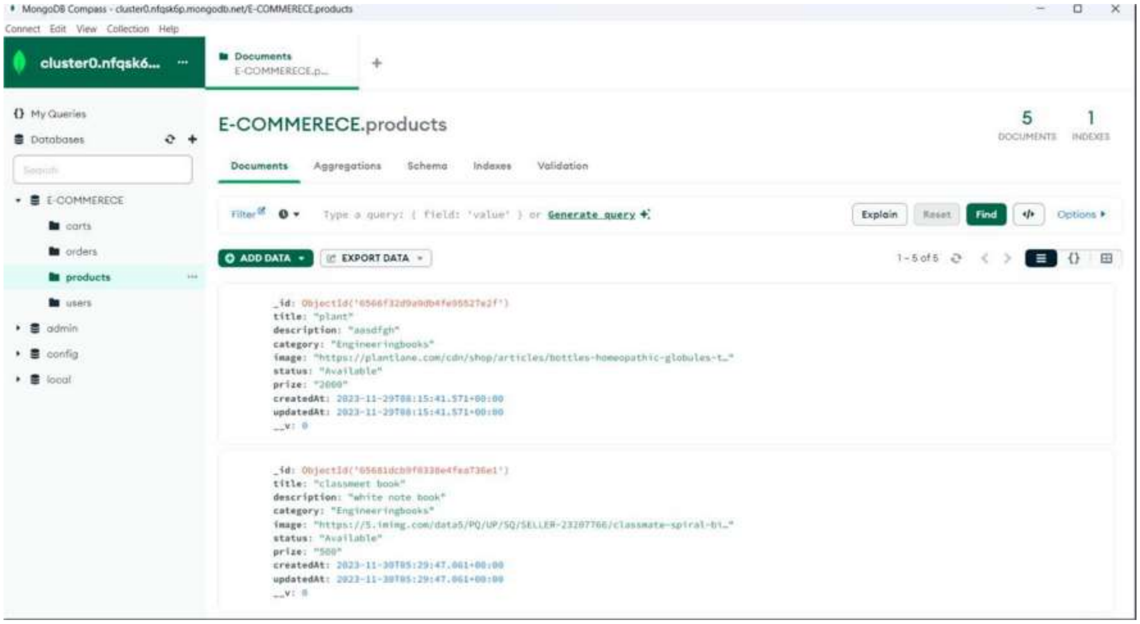


```
_id: ObjectId('6562cc1a175e034ab8b292da')
▸ productid: Array (1)
orderstatus: "order placed"
orderdate: 2023-11-26T04:39:54.985+00:00
user: ObjectId('656221339f5732feaaf4f1c8')
productname: "s 2023-11-26T04:39:54.985+00:00
productimage: "https://i5.walmartimages.com/asr/fa9574f2-2d73-43ba-b3c5-9e0ad2cffe5d_…"
productprice: "1234"
createdAt: 2023-11-26T04:39:54.988+00:00
updatedAt: 2023-11-26T04:39:54.988+00:00
__v: 0

_id: ObjectId('6562f3f744cc7fd62338224c')
▸ productid: Array (1)
orderstatus: "order placed"
orderdate: 2023-11-26T07:29:59.296+00:00
user: ObjectId('6562f32444cc7fd623302142')
productname: "classmeet"
productimage: "https://static1.industrybuying.com/products/office-supplies/notebooks-…"
productprice: "500"
createdAt: 2023-11-26T07:29:59.298+00:00
updatedAt: 2023-11-26T07:29:59.298+00:00
```

Fig 5.4.7

Product Data:

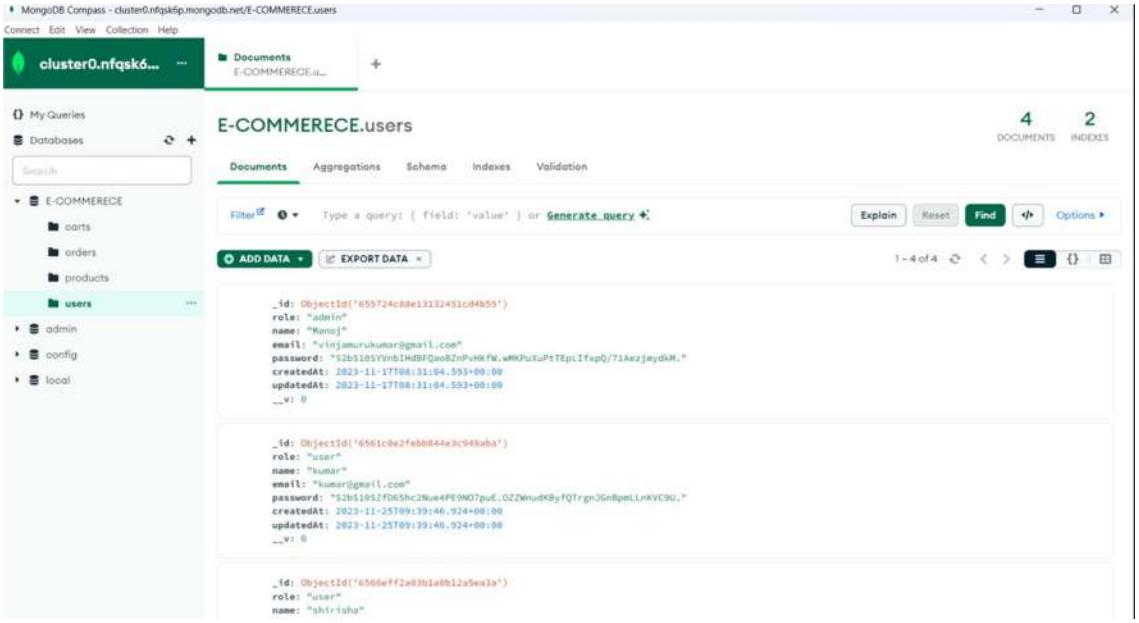


```
_id: ObjectId('6566f32d9a9db4fe9552fe2f')
title: "plant"
description: "aasdfgh"
category: "Engineeringbooks"
image: "https://plantlane.com/cdn/shop/articles/bottles-homeopathic-globules-t…"
status: "Available"
prize: "2000"
createdAt: 2023-11-29T08:15:41.571+00:00
updatedAt: 2023-11-29T08:15:41.571+00:00
__v: 0

_id: ObjectId('65681dcb9f0338e4fea736e1')
title: "classmeet book"
description: "white note book"
category: "Engineeringbooks"
image: "https://5.imimg.com/data5/PQ/UP/SQ/SELLER-23207766/classmate-spiral-bi…"
status: "Available"
prize: "500"
createdAt: 2023-11-30T05:29:47.061+00:00
updatedAt: 2023-11-30T05:29:47.061+00:00
__v: 0
```

Fig 5.4.8

User Data:



```
_id: ObjectId('655724c88e13132451cd4b55')
role: "admin"
name: "Manoj"
email: "vinjamurukumar@gmail.com"
password: "$2b$10$YVnbIHdBFQaoBZnPvHKfW.wMKPuXuPtTEpLIfxpQ/71AezjmydkM."
createdAt: 2023-11-17T08:31:04.593+00:00
updatedAt: 2023-11-17T08:31:04.593+00:00
__v: 0

_id: ObjectId('6561c0e2febb844e3c948aba')
role: "user"
name: "kumar"
email: "kumar@gmail.com"
password: "$2b$10$2fD65hc2Nue4PE9NO7puE.OZZWnudKByfQTrgnJGnBpmLinKVC9U."
createdAt: 2023-11-25T09:39:46.924+00:00
updatedAt: 2023-11-25T09:39:46.924+00:00
__v: 0

_id: ObjectId('6566eff2a03b1a0b12a5ea3a')
role: "user"
name: "shirisha"
```

Fig 5.4.9

Result Analysis:

The result analysis of data storage and transactions in the cloud environment is a crucial aspectthat gauges the effectiveness, efficiency, and overall performance of the implemented system. One key metric is the system's ability to efficiently store and retrieve data in the cloud, ensuringthat the storage mechanisms are not only secure but also scalable to accommodate varying workloads. The analysis involves assessing the responsiveness of the system, measuring how quickly it processes transactions and retrieves data, especially under peak loads and diverse scenarios.

VI. CONCLUSION

So based on this NoSQL database will be proved multi row transaction has been successfully executed for all process. Feature is automatically database reduce its load without admin support

FUTURE ENHANCEMENT:

In the concept multi row transaction using DB2 database has beenmore important one for all the process that scalability process to done the project work. In feature thing is all multi rows. If user at stable mode means database will be remove the userfrom db.

Future enhancements for the analysis of data storage and transactions in the cloud environment could revolve around incorporating cutting-edge technologies and addressing evolving challenges in the dynamic landscape of cloud computing. One avenue for enhancement involves further leveraging artificial intelligence and machine learning algorithms to enhance predictive analytics capabilities. These advancements could optimize resource allocation, predict potential system failures, and automate decision-making processes, contributing to increased efficiency and responsiveness

REFERENCES

[1] F. Chang, J. Dean, S. Ghemawat, W.C. Hsieh, D.A. Wallach,M. Burrows, T. Chandra, A. Fikes, and R.E. Gruber, ‘‘Bigtable: ADistributed Storage System for Structured Data,’’ ACMTrans.Comput. Syst., vol. 26, no. 2, pp. 1-26, June 2008.

[2] B.F. Cooper, R. Ramakrishnan, U. Srivastava, A. Silberstein,P. Bohannon, H.-A. Jacobsen,

[3] N. Puz, D. Weaver, and R. Yerneni,‘‘Pnuts: Yahoo!’s Hosted Data Serving Platform,’’ Proc. VLDBEndowment, vol. 1, no. 2, pp. 1277-1288, Aug. 2008.

[4] Apache, Hbase. [Online]. Available: http://hbase.apache.org/ .

[5] J. Baker, C. Bond, J. Corbett, J.J. Furman, A. Khorlin, J. Larson,J.-M. Leon, Y. Li, A. Lloyd, and V. Yushprakh, ‘‘Megastore:Providing Scalable, Highly Available Storage for InteractiveServices,’’ in Proc. CIDR, 2011, pp. 223-234.

[6] S. Das, D. Agrawal, and A.E. Abbadi, ‘‘G-Store: A Scalable DataStore for Transactional Multi Key Access in the Cloud,’’ in Proc.ACM Symp. Cloud Comput., 2010, pp. 163-174.

[7] T. Haerder and A. Reuter, ‘‘Principles of Transaction-OrientedDatabase Recovery,’’ ACMComput. Survey, vol. 15, no. 4, pp. 287-317, Dec. 1983.

[8] T.P. Council, San Francisco, CA, USATPC-C Benchmark.[Online]. Available: http://www.tpc.org/tpcc .H. Berenson, P. Bernstein, J. Gray, J. Melton, E. O’Neil, andP. O’Neil, ‘‘A Critique of ANSI SQL Isolation Levels,’’ in Proc.ACM SIGMOD, 1995, pp. 1-10.

[9] A. Fekete, D. Liarokapis, E. O’Neil, P. O’Neil, and D. Shasha,‘‘Making Snapshot IsolationSerializable,’’ ACM Trans. DatabaseSyst., vol. 30, no. 2, pp. 492-528, June 2005.

[10] M. Bornea, O. Hodson, S. Elnikety, and A. Fekete, ‘‘One-CopySerializabilityWith Snapshot Isolation Under the Hood,’’ in Proc.IEEE ICDE, Apr. 2011, pp. 625-636.