Madhuri A. D, 2026, 14:2
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

Sync Board: An Intelligent Platform for Real-Time Data Sharing Across Devices

Madhuri A. D[1], Sharadha D[2], Mrudula S[3], Anjali S[4,] Kartik Y[5]

[1]Assistant Professor: Department of Computer Engineering A.C. Patil College of Engineering, Mumbai University Mumbai, India
[2,3,4,5]Students: Department of Computer Engineering A.C. Patil College of Engineering, Mumbai University Mumbai, India

Abstract- Today, many people use more than one device for study, work, or daily tasks. Sharing small information like text or files between devices can be difficult and time-consuming when using emails or messaging apps. To solve this problem, Sync Board is developed as a simple and user-friendly system that allows instant sharing of content a cross device. Sync Board provides a web-based platform where users can add text or upload files on one device and see the same content appear immediately on other connected devices. The system works in real time and does not require complex setup or technical knowledge. Its clean and easy interface makes it suitable for users of all skill levels. The system helps save time, reduce manual effort, and improve productivity. Sync Board is useful for students, professionals, and teams who need quick and smooth content sharing. This project shows how a simple and efficient design can make real-time technology easy to use and accessible for everyone.

Keywords: Real-time synchronization, Cross-device sharing, User-friendly system, Web-based application, Content sharing, Productivity enhancement.

I. INTRODUCTION

In today's digital world, users often work on multiple devices such as laptops, desktops, and mobile phones. While switching between these devices, they frequently need to share text, notes, or files. Manually transferring this information can be time-consuming and disrupt the workflow.

Most traditional sharing methods, such as emails, messaging applications, or cloud storage, require multiple steps and depend on external platforms.

These methods are not suitable for quick data transfer and often reduce efficiency, especially when users need instant access to the same content on different devices. To solve this problem, Sync Board is developed as a simple and user-friendly real-time content synchronization system. It allows users to share text and files instantly across devices through aweb-based platform, eliminating the need for repetitive copy- paste actions or third-party applications. One of the important features of Sync Board is the Switch Environment option. This feature allows users to switch between different working environments or sessions easily. Each environment maintains its own synchronized content, helping users organize data separately for different tasks, projects, or teams.

The system does not require complex setup or technical knowledge, making it easy for anyone to use directly through a web browser. The interface of Sync Board is designed to be clean and simple. Users can upload files, edit text, view shared content, check history, and switch environments smoothly without confusion. This makes the system suitable for both technical and non- technical users.

Overall, Sync Board provides a practical and efficient solution for real-time content sharing across devices. With features like instant synchronization, environment switching, and a user-friendly design, the system improves productivity

II. RELATED WORK

Real-time content sharing and synchronization across evices has gained significant attention with the increased use of multiple digital devices. Users often need to transfer text, files, and notes between systems quickly and efficiently. Traditional sharing methods are not always suitable for real- time collaboration and seamless workflow. As a result, several tools and techniques have been developed to improve cross- device content synchronization. This section discusses existing approaches related to real-time content sharing systems.

A. Manual and Traditional Sharing Methods

Earlier methods of sharing content between devices mainly involved manual techniques such as email attachments, messaging applications, USB drives, or cloud storage services. These methods require multiple steps and often interrupt the user's workflow.

Although these techniques are widely used, they are not efficient for frequent or real-time data sharing. Users must repeatedly upload, download, and manage files, which increases time consumption and reduces productivity. These limitations led to the need for faster and more automated content synchronization solutions.

B. Cloud-Based Storage and Synchronization Systems

Cloud-based platforms such as online storage services provide automatic synchronization of files across devices. These systems allow users to access the same data from multiple devices using internet connectivity.

While cloud-based solutions offer convenience, they often require user accounts, authentication, and storage management. Additionally, synchronization delays may occur, and these systems are not optimized for instant sharing of small pieces of data like copied text or temporary notes.

C. Real-Time Communication-Based Systems

With the advancement of real-time web technologies, systems using WebSocket's and real-time communication protocols have been developed. These systems enable instant data transfer between connected devices without delay.

Real-time communication-based systems are effective for live collaboration and instant updates. However, many existing solutions lack a simple user interface or require technical configuration, which can be challenging for non-technical users.

D. Web-Based Collaborative Content Sharing Platforms

Recent web-based platforms focus on collaborative content sharing by allowing multiple users to view and edit shared content simultaneously. These systems support real-time updates and improve teamwork efficiency.

Although such platforms offer powerful features, they are often complex and overloaded with functionalities. This complexity makes them less suitable for users who only need a simple and quick content synchronization tool. Sync Board addresses this gap by providing a lightweight, user-friendly, and real-time solution focused on ease of use and efficiency.

III. PROPOSED SYSTEM

The proposed system, Sync Board – Real-Time Cross-Device Content Synchronization System, is designed to simplify the process of sharing text and files across multiple devices instantly. The system eliminates the need for traditional methods such as emails, messaging apps, or cloud uploads, which are time-consuming and interrupt workflow. Sync Board provides a seamless, real-time platform for content synchronization.

The main objective of Sync Board is to offer a simple and user- friendly solution that allows users to share copied text and uploaded files across devices in real time. The system ensures that content added from one device is immediately available on all other connected devices without delay. Sync Board follows a modular architecture where each module performs a specific function such as data input, real-time synchronization, environment switching, and content display. These modules work together to provide a smooth and efficient user experience.

A. System Overview

Sync Board is a web-based real-time synchronization system that enables instant sharing of content between devices. Users can create or join an environment, add text or files, and view the same content across all devices connected to that environment.

The system focuses on solving common problems such as repetitive copy-paste actions, slow file transfers, and dependency on third-party applications. By using real-time communication technology, Sync Board ensures fast and reliable content synchronization with minimal user effort.

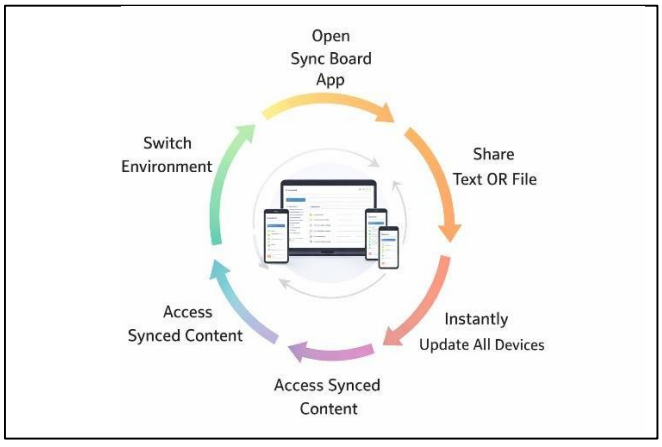

Fig. 1. System Overview

B. System Architecture

1. The architecture of Sync Board consists of multiple interconnected modules that work together to achieve Realtime synchronization. These modules include the User Interface Module, Environment Management Module, Real-Time Sync Engine, Data Storage Module, and Output Module.

2. The User Interface Module provides a clean and simple interface where users can add text, upload files, view shared content, and switch between environments easily.

3. The Environment Management Module allows users to create, join, and switch between different environments. Each environment maintains its own synchronized content, enabling better organization for different tasks or projects.

4. The Real-Time Sync Engine is responsible for instantly transferring content updates across all connected devices. It ensures that any change made on one device is immediately reflected on others.

5. The Data Storage Module stores shared content securely so that it can be accessed by users connected to the same environment at any time.

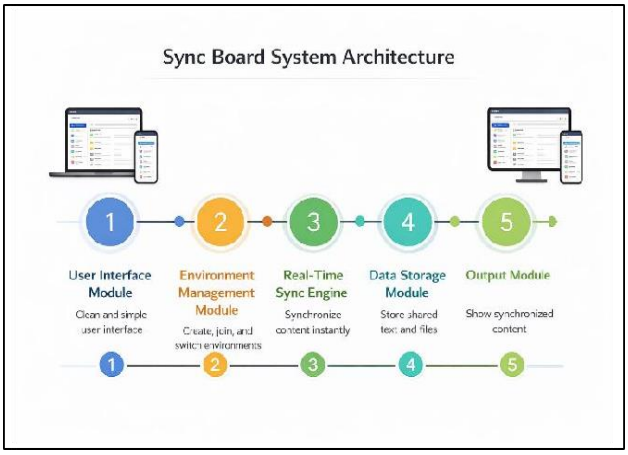

Fig. 2. System Architecture

A. Functional Workflow

1. The process starts when the user opens the Sync Board web application.

2. The user enters text or uploads a file into the application.
3. The client device sends the entered data to the central server.
4. The server processes the received data and updates the shared content.
5. The server then broadcasts the updated data to all connected devices in real time.
6. Other devices connected to the same environment receive the updated content instantly.
7. The synchronized content is displayed on all devices simultaneously.
8. The process ends after successful synchronization.

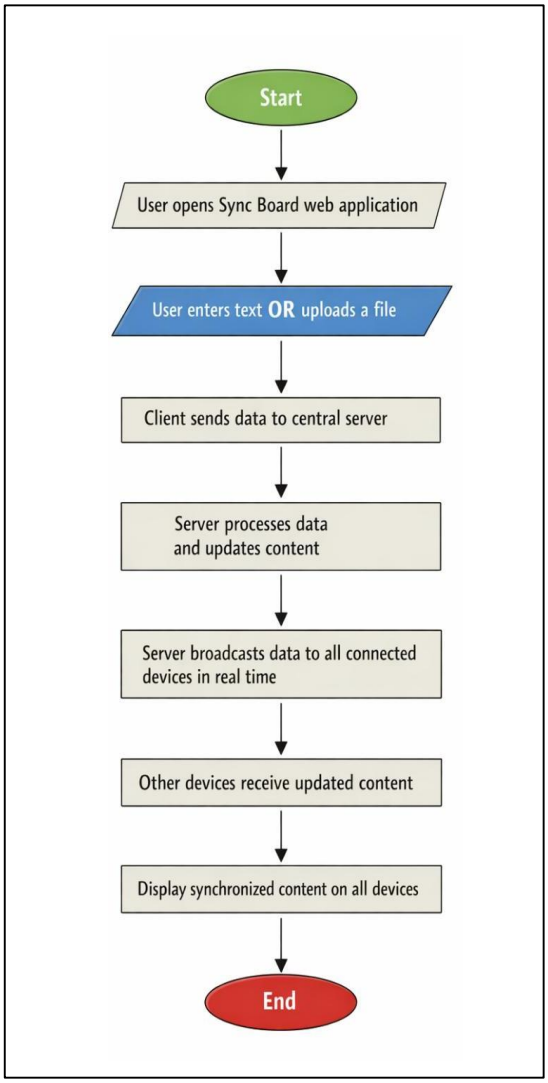

Fig. 3. Functional workflow of the Sync Board system

B. Features and Advantages

Real-Time Content Synchronization:
Sync Board instantly synchronizes text and files across all connected devices, ensuring that updates appear in real time without manual refreshing.

Multi-Device Accessibility:
The system allows users to access synchronized content on multiple devices such as laptops, tablets, and smartphones using a web browser.

Environment-Based Sharing:
Users can create, join, and switch between different environments, enabling organized and controlled sharing of content for different groups or tasks.

User-Friendly Interface:
Sync Board provides a clean and intuitive interface that allows users to easily enter text, upload files, and view synchronized content without technical knowledge.

Fast and Efficient Communication:
The system uses a centralized server to process and broadcast updates quickly, reducing delays and improving collaboration efficiency.

Secure and Controlled Access:
Only users connected to the same environment can view or modify shared content, ensuring basic privacy and controlled data access.

Time-Saving Collaboration Tool:
By eliminating the need for manual copying or repeated file transfers, Sync Board significantly reduces time and effort in content sharing.

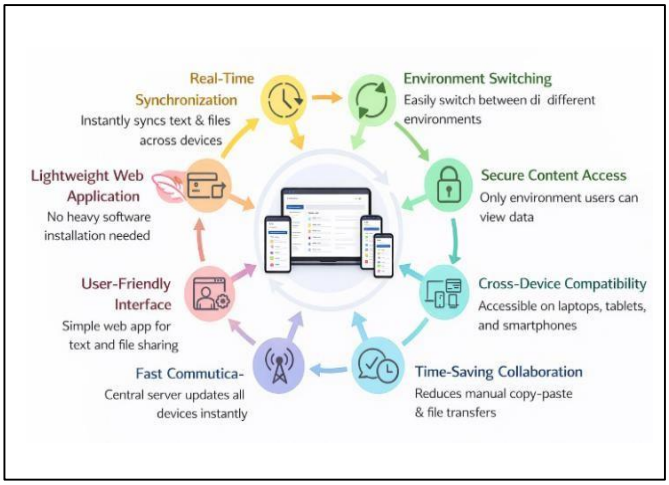

Fig 4: Benefits

IV. Methodology

The methodology of the **Sync Board** system focuses on designing and implementing a real-time content synchronization platform that allows users to share and access information seamlessly across multiple devices. The system follows a structured approach that includes environment creation, data input, real-time synchronization, conflict handling, and content visualization. Each stage plays a crucial role in ensuring smooth, fast, and reliable data sharing
The methodology is divided into the following stages:

A. Environment Creation and User Setup

The first step in the Sync Board workflow is environment creation. Users create or join a specific environment that acts as a shared workspace. Each environment allows multiple users to collaborate by accessing the same synchronized content.

User authentication ensures secure access to environments. Once logged in, users can select or switch between environments based on their requirements, enabling organized and controlled collaboration.

B. Input and Content Capture

After entering an environment, users can input data in the form of text, notes, or files. The system captures the content entered by users through a user-friendly interface designed for quick and easy interaction.

All input data is tagged with environment-specific identifiers to ensure that content is shared only among users within the same environment.

C. Data Processing and Synchronization Engine

The core component of Sync Board is the synchronization engine. This module processes incoming data and prepares it for real-time sharing. The system ensures that updates made by one user are instantly reflected across all connected devices within the same environment. Efficient data handling mechanisms are used to minimize delay and ensure smooth synchronization without data loss minimize delay and ensure smooth synchronization without data loss.

D. Real-Time Update and Conflict Handling

During real-time collaboration, multiple users may update content simultaneously. The system continuously monitors such actions and handles synchronization conflicts effectively.

If conflicting updates occur, Sync Board applies predefined rules to manage and merge changes, ensuring consistency across all devices and preventing data overwrite issues.

E. Content Storage and Management

All synchronized data is securely stored in a centralized database. The system maintains version control to track updates and allow recovery if needed.

This structured storage approach ensures data reliability, scalability, and long-term accessibility across multiple environments.

F. Output Display and Visualization

The synchronized content is displayed in real time on all user devices in a clean and organized format. Users can view updates instantly without refreshing the system.

The interface supports easy content visualization, making Sync Board suitable for classrooms, meetings, and collaborative workspaces.

G. System Workflow Summary

The overall methodology ensures efficient, secure, and real-time content synchronization. By combining environment-based collaboration, real-time data processing, and conflict handling mechanisms, the Sync Board system provides a reliable and user-friendly solution for multi-device content sharing. H. WebSocket-Based Real-Time Synchronization Algorithm

H. Web Socket Based Real Time Synchronization Algorithm

Algorithm: WebSocket-Based Real-Time Synchronization in Sync Board

Algorithm Objective:

To achieve real-time synchronization of text and file data across multiple devices using a persistent WebSocket connection and environment-based communication.

Algorithm Description:

Sync Board uses a WebSocket-based full-duplex communication model where each environment is mapped to a logical server-side channel. Updates from one device are instantly broadcast to all devices connected to the same environment.

Step-by-Step Technical Algorithm:

Step 1: WebSocket Initialization -

The client application initializes and sends a WebSocket connection request to the Sync Board server.

Step 2: Persistent Connection Establishment -

The server accepts the WebSocket handshake and establishes persistent, bidirectional communication channel.

Step 3: Environment Mapping -

The client sends an Environment_ID.
The server binds the client socket to the corresponding environment channel.

Step 4: Event Detection -

The client continuously monitors events such as:

- Text updates
- File upload actions

Step 5: Data Serialization and Transmission -

On detecting a change, the client serializes the updated content into JSON format and sends it to the server via WebSocket.

Step 6: Server Processing and Broadcasting -

The server validates the data, updates the environment state, and broadcasts the update to all sockets mapped to the same Environment_ID

Step 7: Client Rendering -

Receiving clients deserialize the data and instantly update the live editor and shared file view.

V. Results and Discussion

The performance evaluation of the **Sync Board** system was conducted to assess its effectiveness in real-time content synchronization across multiple devices. The system was tested using sample textual data shared between multiple users connected simultaneously. The results demonstrate that the system successfully synchronized content in real time with minimal delay while maintaining data consistency across all connected devices.
The automated synchronization mechanism significantly reduced manual effort required for sharing information between devices and eliminated inconsistencies caused by delayed updates. The system proved to be reliable, fast, and suitable for collaborative use cases such as group discussions, classrooms, and remote teamwork.

Key features evaluated included:

1. Live content updates
2. Clear text display across devices
3. Immediate reflection of user modifications
4. Minimal latency during synchronization

User feedback indicated that the interface was intuitive and easy to use, even for first-time users. The real-time visualization significantly enhanced collaborative productivity.

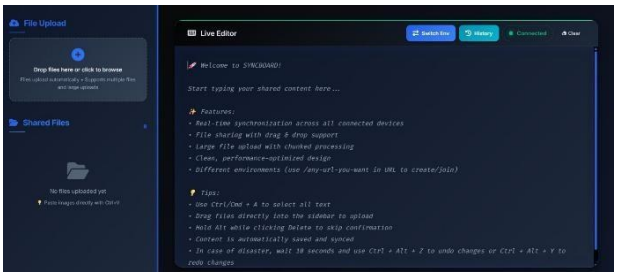

Fig 5: Output window
Dashboard of Sync Board

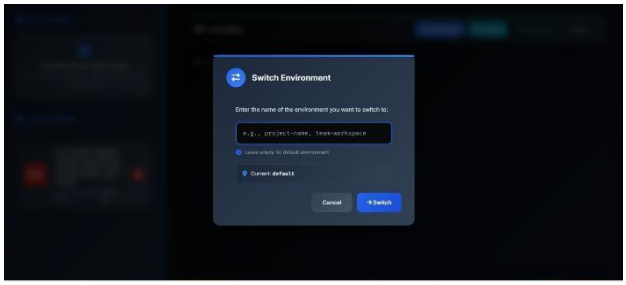

Fig 6: Environment management window

"Environment switching dialog in Sync Board used to select or change the working environment."

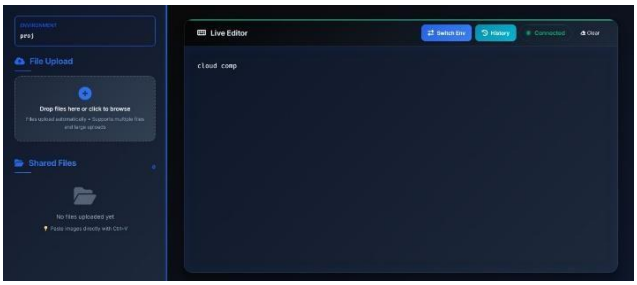

Fig 7: Main user interface of the Sync Board system

"This figure shows the main interface of the Sync Board application with a live editor, file upload, shared files panel, and environment selection for instant collaboration."

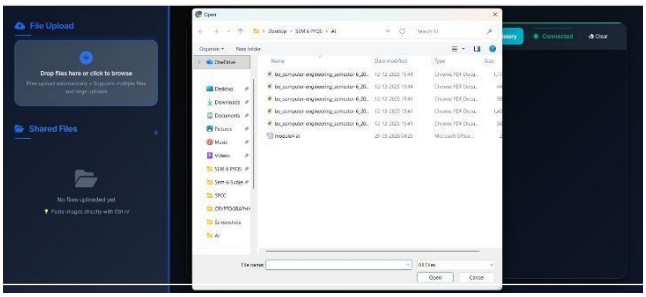

Fig 8: Interface for uploading and sharing files

"This feature lets users upload files and share them instantly with others in the same environment. No login or registration is required"

Fig. 9: File Upload and Display in Sync Board

"This figure shows the file upload feature of Sync Board, where a user uploads a file and the uploaded file is instantly displayed to all connected users in the same environment."

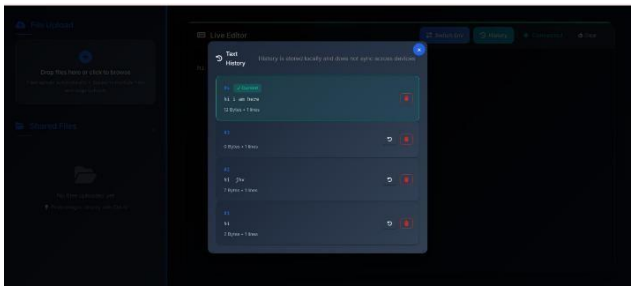

Fig 10: Text history window in Sync Board

"Text history panel showing previously shared content in Sync Board.

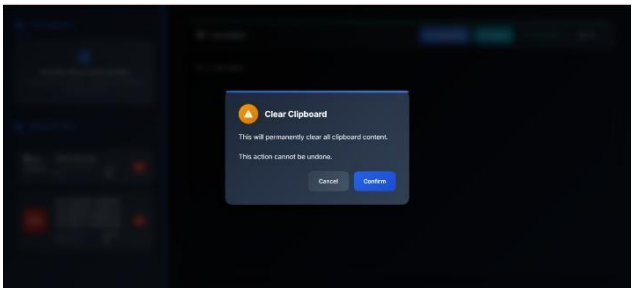

Fig. 11: Clear Clipboard Function

"This figure shows the clear clipboard feature in Sync Board, which allows users to remove all shared text and files from the current environment instantly."

F. Discussion

The results confirm that the Sync Board system effectively addresses common challenges in real-time content sharing, such as delayed updates, inconsistent data, and manual information synchronization mechanism ensured fast and reliable data propagation while maintaining consistency across all connected devices.

on stable connectivity for optimal performance, testing showed that synchronization remained reliable under normal usage conditions. Minor delays were observed only during simultaneous high-frequency updates, which can be further optimized in future enhancements.

Overall, the Sync Board system provides an efficient, scalable, and user-friendly solution for real-time collaborative content synchronization. The system reduces manual workload, improves communication speed, and offers a practical solution suitable for educational and collaborative environments.

VI. Conclusion and Future Work

This study presented Sync Board, a real-time content synchronization system designed to simplify and enhance collaborative information sharing across multiple devices. The system was developed to address common challenges in traditional content sharing methods, such as delayed updates, data inconsistency, and the manual effort required to transfer information between devices. By implementing a real-time synchronization mechanism and centralized data handling, Sync Board successfully enabled instant and consistent content sharing.

Experimental evaluation demonstrated that the system efficiently handled simultaneous user updates while maintaining data consistency and minimal synchronization delay. The real-time update mechanism ensured that changes made by one user

T The implementation of Sync Board highlights the effectiveness of real-time synchronization systems in improving collaboration and communication efficiency. The results indicate that such systems can significantly reduce manual effort, minimize errors caused by outdated information, and support seamless collaboration in academic, professional, and group-based environments.

Future Work

Future enhancements of the Sync Board system can focus on the following areas:

- Improve real-time synchronization performance.
- Simplify environment management and switching.
- Add offline update recovery after reconnection.
- Enhance user interface and usability.
- Support higher scalability

REFERENCES

1. G. Coulouris, J. Dollimore, T. Kindberg, and G. Blair, Distributed Systems: Concepts and Design, 5th ed. Pearson Education, 2012.
2. A. S. Tanenbaum and M. van Steen, Distributed Systems: Principles and Paradigms, 2nd ed. Prentice Hall, 2007.
3. D. P. Anderson, "Real-time collaborative systems and synchronization challenges," IEEE Internet Computing, vol. 18, no. 4, pp. 72–76, Jul. 2014.
4. S. R. Ellis, C. Gibbs, and G. Rein, "Groupware: Some issues and experiences," Communications of the ACM, vol. 34, no. 1, pp.39-58, Jan. 1991.
5. M. Pimentel and B. G. Nickerson, "Communicating and displaying real-time collaborative data," Computer Networks, vol. 31, no. 11–16, pp. 1507–1521, 1999.
6. H. Li and M. Chen, "Design and implementation of real-time data synchronization systems," International Journal of Computer Applications, vol. 45, no. 7, pp. 22–27, 2012.
7. E. Gamma, R. Helm, R. Johnson, and J. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 1994.
8. I. Sommerville, Software Engineering, 10th ed. Pearson, 2016.
9. M. Fielding and J. Reschke, Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content, IETF RFC 7231, Jun. 2014.
10. T. Berners-Lee, R. Fielding, and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax," IETF RFC 3986, Jan. 2005.
11. T. Berners-Lee, R. Fielding, and L. Masinter, "Uniform

Resource Identifier (URI): Generic Syntax," IETF RFC 3986, Jan. 2005.

12. P. Bellavista, A. Corradi, and C. Stefanelli, "Mobile agent middleware for mobile computing," Computer, vol. 34, no. 3, pp. 73–81, Mar. 2001.
13. M. Kleppmann, Designing Data-Intensive Applications, O'Reilly Media, 2017
14. J. Waldo, G. Wyant, A. Wollrath, and S. Kendall, "A note on distributed computing," Sun Microsystems Laboratories, Tech. Rep., 1994.
15. R. Fielding, "Architectural styles and the design of network-based software architectures," Ph.D. dissertation, Univ. of California, Irvine, 2000.
16. F. Dabek, R. Cox, M. Kaashoek, and R. Morris, "Vivaldi: A decentralized network coordinate system," ACM SIGCOMM Computer Communication Review, vol. 34, no. 4, pp. 15–26, Aug. 2004.
17. J. Han, E. Haihong, G. Le, and J. Du, "Survey on NoSQL database," Proc. 6th Int. Conf. Pervasive Computing and Applications, pp. 363–366, 2011.
18. A. Banks and R. Gupta, "MQTT Version 3.1.1," OASIS Standard, Oct. 2014.
19. S. Tilkov and S. Vinoski, "Node.js: Using JavaScript to build high-performance network programs," IEEE Internet Computing, vol. 14, no. 6, pp. 80–83, Nov.–Dec. 2010.
20. K. Birman, "The process group approach to reliable d distributed computing," Communications of the ACM, vol. 36, no. 12, pp. 36–53, Dec. 1993
21. J. Dean and L. A. Barroso, "The tail at scale," Communications of the ACM, vol. 56, no. 2, pp. 74–80, Feb. 2013.
22. M. P. Papazoglou, "Service-oriented computing: Concepts, characteristics and directions," Proc. Int. Conf. Web Information Systems Engineering, pp. 3–12, 2003.
23. S. Sakr, A. Liu, and A. Batista, "A survey of large scale data management approaches in cloud environments," IEEE Communications Surveys & Tutorials, vol. 13, no. 3, pp. 311–336, 2011.
24. D. Taibi and V. Lenarduzzi, "On the definition of microservice bad smells," IEEE Software, vol. 35, no. 3, pp. 56–62, May–Jun. 2018.