Design and Development of Billing Module for DRM

Vaishali Shirsath[1]
Department of Information Technology Vidyavardhini's College of Engineering and Technology Vasai, India

Pragati Patil[2]
Department of Information Technology Vidyavardhini's College of Engineering and Technology Vasai, India

Manaswi Jadhav[3]
Department of Information Technology Vidyavardhini's College of Engineering and Technology Vasai, India

Omkar Jadhav[4]
Department of Information Technology Vidyavardhini's College of Engineering and Technology Vasai, India

Sanskruti Kokare[5]
Department of Information Technology Vidyavardhini's College of Engineering and Technology Vasai, India

Dr. Shreya Bokare[6]
Centre for Development of Advanced Computing C-DAC Mumbai, India

Abstract:- The goal of the proposed project is to develop a billing system that is both scalable and adaptable to meet the specific needs of data replication services. Service providers will be able to accurately bill their clients using this billing module's many capabilities, which include usage-based pricing, multi tiered billing structures, and real-time cost tracking. For the Billing Module, we will create an adaptable and expandable system architecture. varied storage technologies, including cloud based and on-premises solutions, as well as varied data replication techniques, including synchronous and asynchronous replication, will be supported by the architecture. The intricacies of data replication utilisation can be monitored using the invoicing module. This covers the amount of data involved, the frequency of replication, and the number of source and target systems. To provide accurate bills for clients, the consumption data will be gathered and processed. The billing module will handle multi tiered charging structures in order to serve a wide range of customers. It is possible to design various customer segments or pricing plans, each with a unique pricing structure and set of billing specifications. The development of an intuitive web based dashboard will enable users to view their bills, expenses, and use. In addition, historical data and budget threshold setting will be available on this dashboard. Service providers will be able to implement charging capabilities with minimal disruption because to the billing module's seamless integration with current data replication systems. Customization choices will also be offered to companies with certain billing requirements.

Keywords:- Scalable, Adaptable, Synchronous, Asynchronous Replication, Customization.

I. INTRODUCTION

In Disaster Recovery Management [4], replication of data poses vital role. After any disaster or data breaches occur, it is necessary to move or replicate and store the data from primary site to secondary site. This data replication is mainly took place by three methods i.e., Block Level replication, File Level replication and MongoDB replication.

Previously for such transactions were done manually also the automated invoice generation was not available. Hence the development of billing module for data replication process is able to generate bills for each user depending upon the repli- cation method. Rates may differ according to the replication method and duration of transfer [21]. With this billing module user will be able to generate bill and automated invoice for whole transaction.

II. MOTIVATION

The motivation behind this work is to address challenges in current billing systems for distributed computing [24]. The goal is to create a transparent and user-friendly billing system that accurately measures resource usage, empowers users to manage costs, enhances efficiency for service providers, adapts to dynamic computing environments, and ensures compliance with industry standards, ultimately providing a competitive edge in the cloud computing landscape [13].

➢ *Fair Resource Utilization Measurement*
Traditional billing systems may not accurately capture the dynamic resource utilization patterns in distributed computing environments [19]. A motivation for this work is to devise a system that ensures fair and precise measurement of resource usage across various nodes and tasks [6].

➢ *Transparent and Equitable Billing Practices*

Current billing systems [9] might lack transparency, making it challenging for both service providers and customers to understand and validate charges. The motivation here is to es- tablish a billing system that promotes transparency, providing a clear breakdown of costs associated with each task [5].

➢ *Adaptation to Dynamic Computing Environments*

The dynamic nature of distributed computing requires billing systems to adapt swiftly to changes in resource usage [11]. The motivation for this work is to design a billing system capable of dynamically adjusting to fluctuations in computing demands, ensuring accuracy and efficiency in cost calculations.

➢ *Problem Domain*

The problem domain for this work involves the inefficien- cies and challenges present in current billing

systems for distributed computing environment [1]. Key issues include in- accurate measurement of resource usage, lack of transparency in billing practices, user difficulties in cost management, and the need for billing systems to adapt to the dynamic nature of distributed environments [10]. The goal is to address these challenges and create a more efficient, transparent, and adaptable billing system that benefits both service providers and users in the distributed computing ecosystem.

➢ *Problem Representation*

In the realm of distributed computing environments, ex- isting billing systems face inefficiencies and challenges [3]. These encompass inaccuracies in resource usage measurement, opaque billing practices, user complexities in managing costs, and the crucial necessity for billing systems to dynamically adapt to the distributed environment's fluid nature.
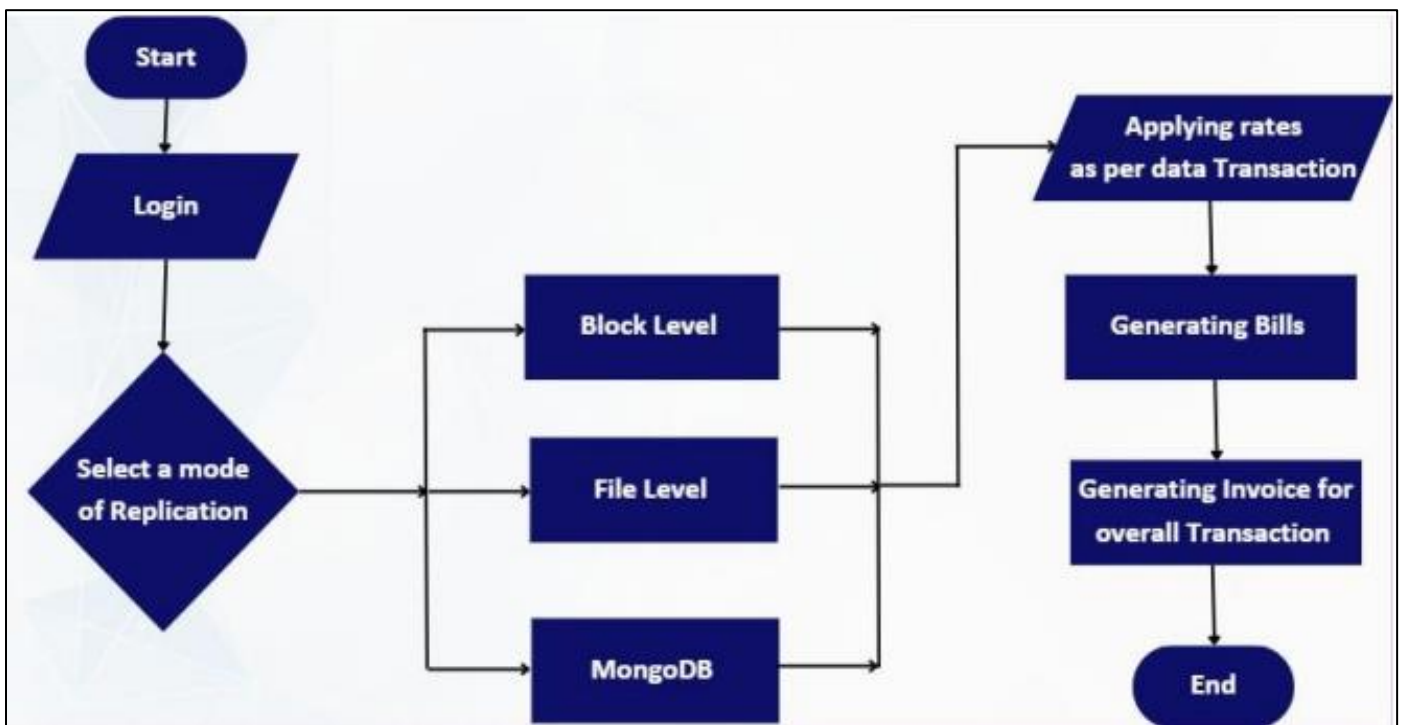

Fig 1 Flowchart (System Outline)

III. METHODOLOGY

➢ *User Interface for Billing Module*

Billing module [15] will be one of the modules of existing DRM system, which will be responsible for making bills for entire transaction of data and also generating the automated invoice. For this we created a user friendly user interface which will help user to easily create their own bills based on their mode of data transaction [16]. This user interface will contain dynamic forms with multiple drop-down menus, which will enable user to select his replication setup ID, start date, end date, technique (active-active or active-passive) and type of database, for data transaction. For replication types certain range is assigned for each replication type which is as described.

- File-level Replication: This replication method operates on individual files. For optimal performance, we recom- mend file sizes ranging from 1 MB to 100 MB.
- Block-level Replication: Block-level replication focuses on smaller units within files, enhancing efficiency. Ideal block sizes typically fall between 1 MB (or 512 KB) and 4 MB.
- MongoDB Replication: MongoDB replication involves replicating data at the document level. To ensure smooth replication, document sizes should ideally range from 1 MB (or 100 KB) to 16 MB.

This all data will get fetched from the database of the DRM system.

➢ *Database of DRM*

Database of DRM system has records of each user who opted for replication of data. [18] This data will be in the form of tables which are created using PostgreSQL. This database typically consists of various tables, which are accessible from user interface via various inner and outer joins among themselves.

➢ *Billing user Details*

Billing user details is one of the most crucial table from the database [8]. It contains user details like user name, application name, technique, direction and replication setup ID. User name field provides name or identity of the user, application field provides type of the database selected, technique field is all about the technique selected for data replication i.e., either active-active or active-passive, direction of the data flow and replication setup ID against each user [22].

➢ *Rate*

Rate is an important table of database, which is responsible for actual calculation of bill [17]. The rate of data per byte is decided according to the type of database. It consists of fields like start time, end time, replication setup ID, application name, user name, charge for one byte, total number of bytes and total amount to pay. In this table, the rate per byte of data will get multiplied with total number of bytes, and will provide total amount to pay by user. For this it requires start time, a time when user started transaction of data and end time, which will be a time when user going to pay a bill [20].

➢ *Payment Details for Invoice Generation*

This particular table in the database will be useful in gener- ating invoice. It consists of fields like user details, application name, start date, end date, amount, time period, status, invoice ID and invoice status [2]. The time period field will show total number days from start date to end date, while status will show whether replication of data is stopped or still it is going on, invoice ID will provide unique ID for each invoice and invoice status will provide information about whether bill is paid or not [20].

IV. RESULTS AND SENSITIVITY ANALYSIS

A number of benefits have resulted from the Data Repli- cation System's Billing Module's responsive user interface (UI) implementation, including improved accessibility and an overall better user experience across various devices [14].

➢ *Enhanced user Context*

Without sacrificing readability or functionality, users may easily use the Billing Module from a range of devices, includ- ing computers, tablets, and smartphones [23]. Interacting with the system is now simple and easy thanks to the touch-friendly design features, adaptable forms, and fluid grid structure.

➢ *Improved Availability*

The Billing Module's responsive user interface (UI) guar- antees that people with disabilities may access and use it with ease, as it conforms to WCAG accessibility criteria [12]. Accessibility has been enhanced for all users with the help of keyboard navigation support, appropriate labelling, and text equivalents for images.

➢ *Adaptive Layouts*

The responsive design has effectively adjusted grid layouts, font sizes, and spacing to fit various screen sizes. Navigation is made simpler by the navigation menu, which changes from a horizontal menu on bigger screens to a collapsible menu on smaller devices.

➢ *Charts and Tables that Respond*

In order to preserve data readability and usefulness on smaller devices, data tables and charts are now responsive [7]. For large tables, using horizontal scrolling helps to preserve data integrity by preventing data truncation.

- Registration page consists of fields like username, email and password. These fields take input from users in alphabetic and alphanumeric system.

Fig 2 Registration Page

- Users can register to the system by using valid username, email address and password. This data will get stored in system's database, using which user will be able to login to the system.
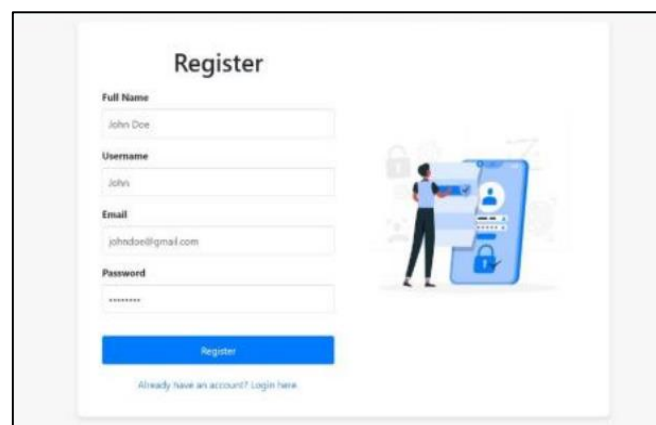

Fig 3 Registration Page with Inputs

- Users can login using login credentials created while registering to the system. Login credentials that required are username and password.

- After logging into the system welcome page will be displayed. This page is containing information regrading about Billing Module. It contain a button at the bottom called "choose your plan", which redirect to the Billing Plan Page for selecting Plans.
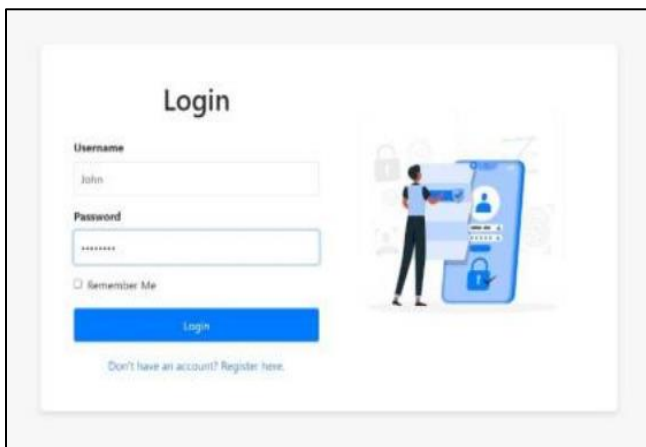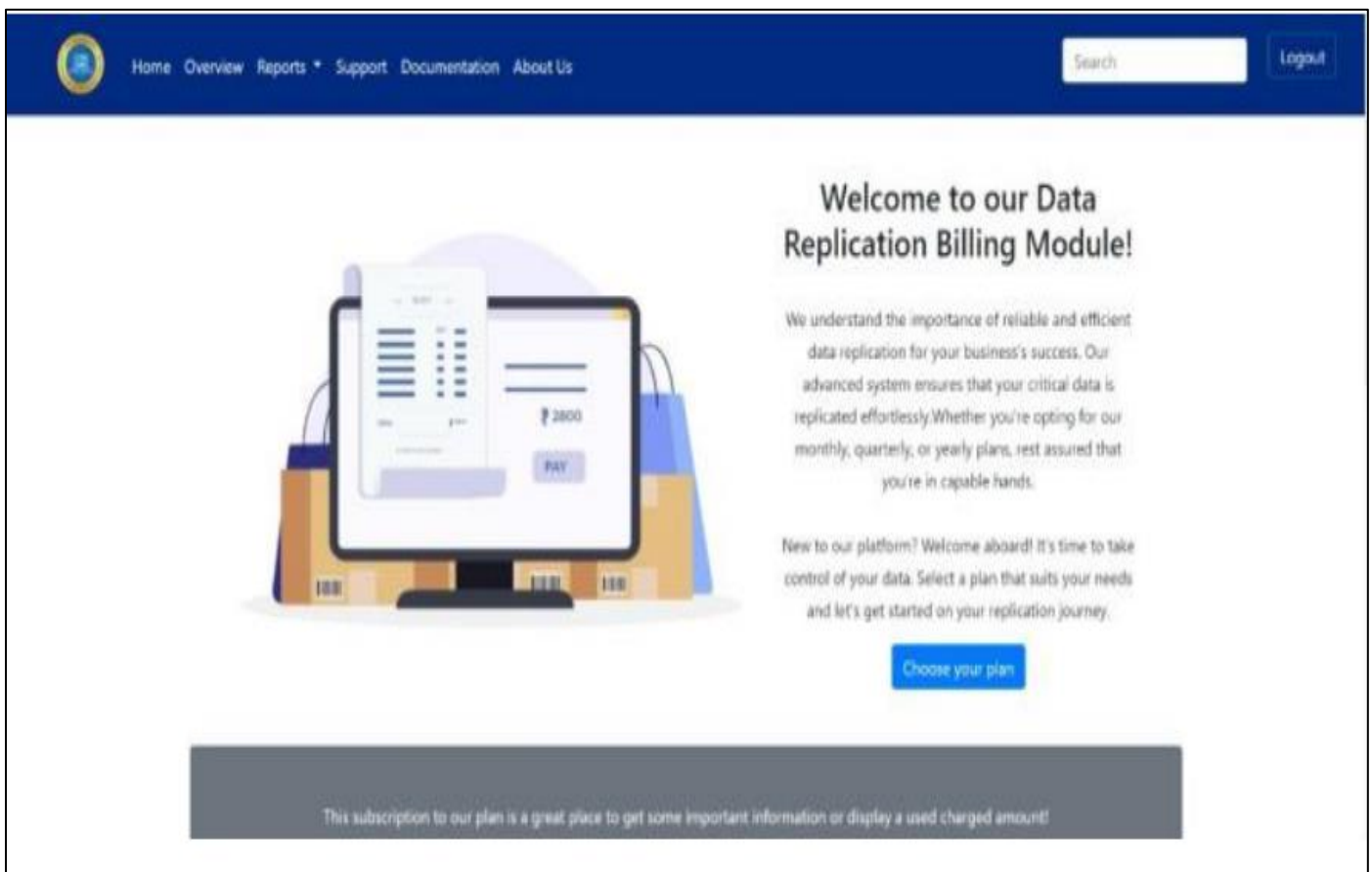

Fig 4 Login Page

Fig 5 Welcome Page

- User have to choose plans based on their duration (time period). Here plans are divided into three type i.e Monthly, Quarterly, Annually. at the end of every container of plans "Get Started" button is present.

Fig 6 Billing Plan Page

- After calculation of bill, automated invoice for each individual user will be generated.This invoice will contain all transactions that are performed by user.This invoice can be saved in the form of PDF file. Users can pay bill using pay now option, which will direct them towards payment gateway.

Fig 7 Invoice Page

- As user select the pay now option, payment gateway page will be opened, which consists of preferred payment methods such as Card, Net banking, Wallet, Pay Later.
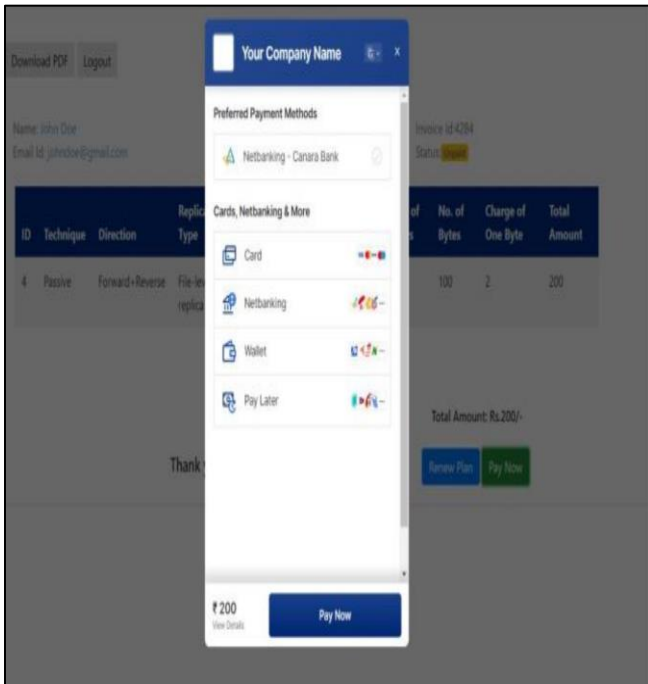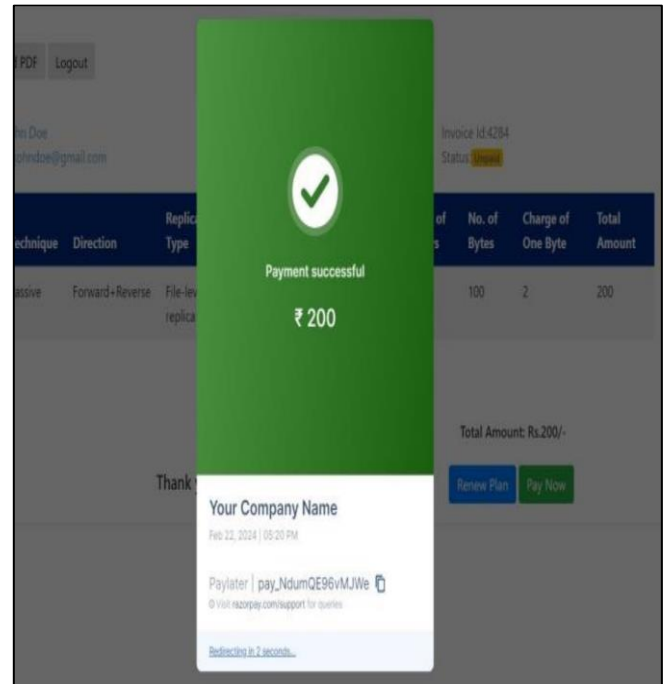
Fig 8 Payment Gateway Page

- User have to select type of the preferred method. After selecting type of preferred method, user will be notified with "Please wait while we redirect you to your bank pages". After filling all details user can make payment by clicking on Submit Payment button.

Fig 9 Payment Gateway Page Loading Page

- After processing of payment, as soon as payment get successful,the message of "Payment Done Successfully", will be displayed on the screen.

Fig 10 Payment Done

- After processing of payment, as soon as payment get successful,the message of "Payment Successful! Payment ID of that Particular user will be displayed against that Payment", will be displayed on the screen.
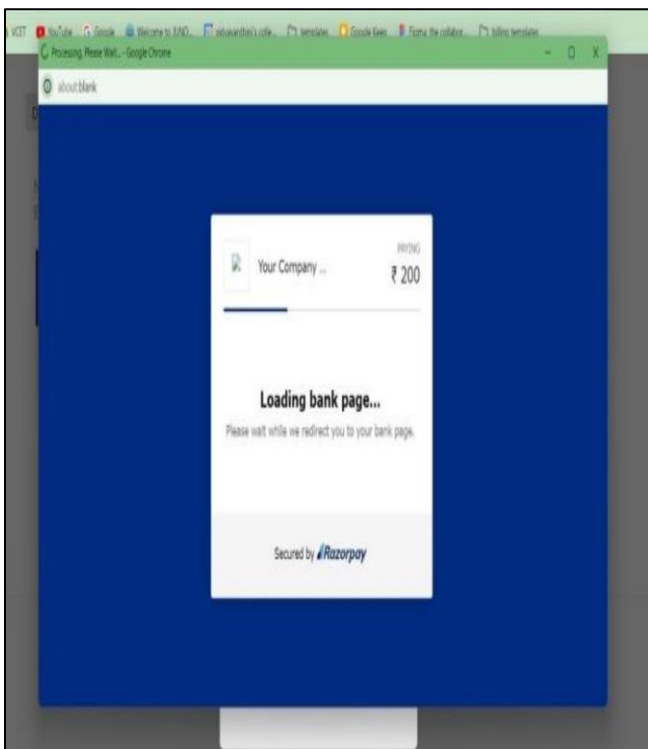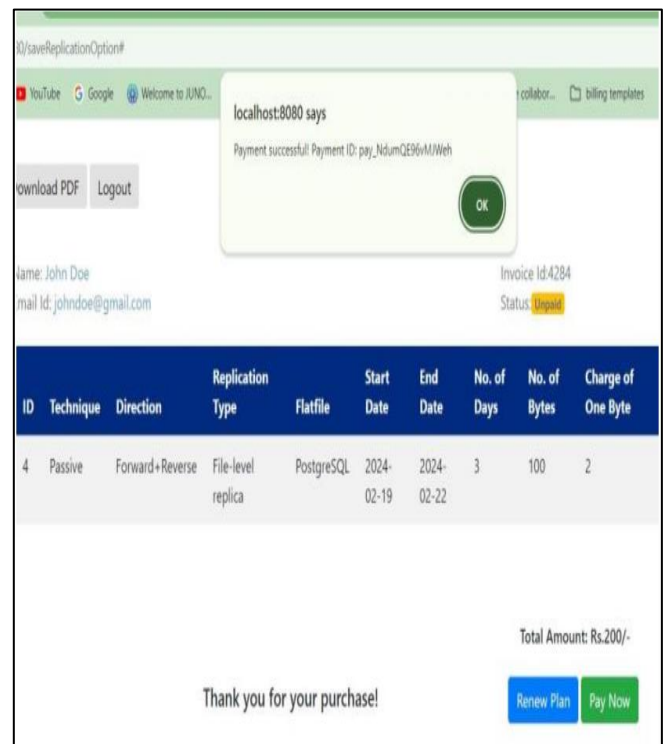

Fig 11 Payment Done in DRM Screen

- Above graph describes the distribution of replication types selected by users.According to the analysis, users mostly opt for the file level replication. MongoDB level replication has lowest users among all three types of replication. As shown in Fig 12.
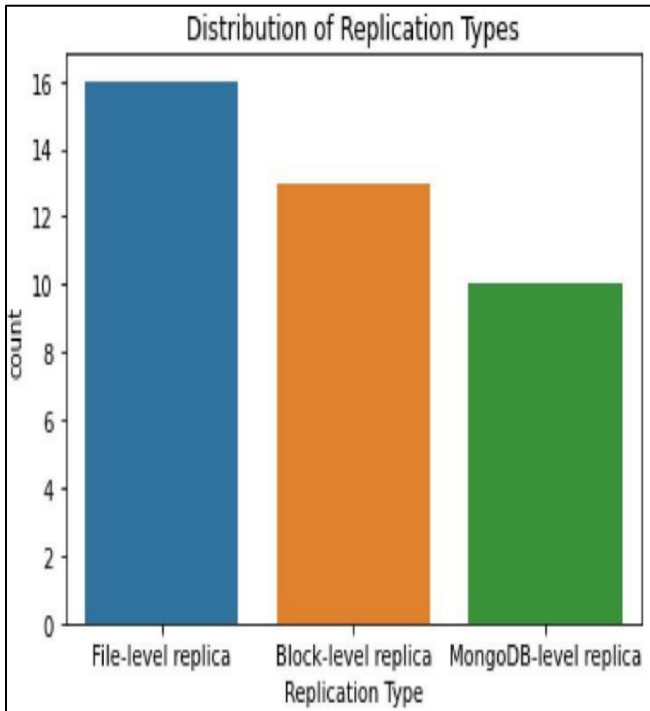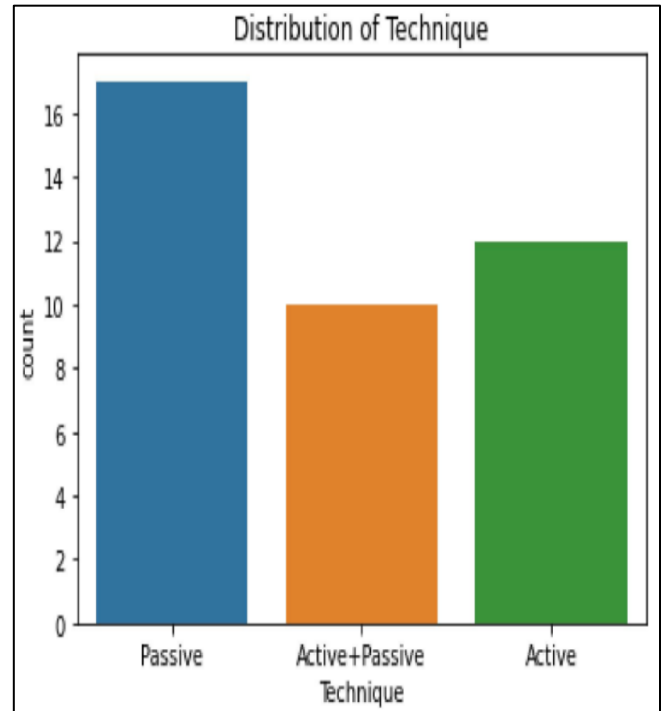
Fig 12 Analysis of Replication Types

- Above graph determines distribution of direction of replication chosen by users among three categories i.e., Forward+Reverse, Forward and Reverse. According to the analysis, most of the users are opting for replication in reverse direction. As shown in Fig 13.

Fig 13 Analysis of Direction of Replication

- This graph shows distribution of techniques if replication used by users. Techniques used for replication are Active, Active-Passive and Passive. As per the analysis, the technique mostly used for replication is passive technique. As shown in Fig 14.

Fig 14 Analysis of Techniques of Replication

- Above figure determines the accuracy of model,in which type of replication, direction of replication and techniques used for replication are major attributes. The accuracy for model is 0.75. As shown in Fig 15.
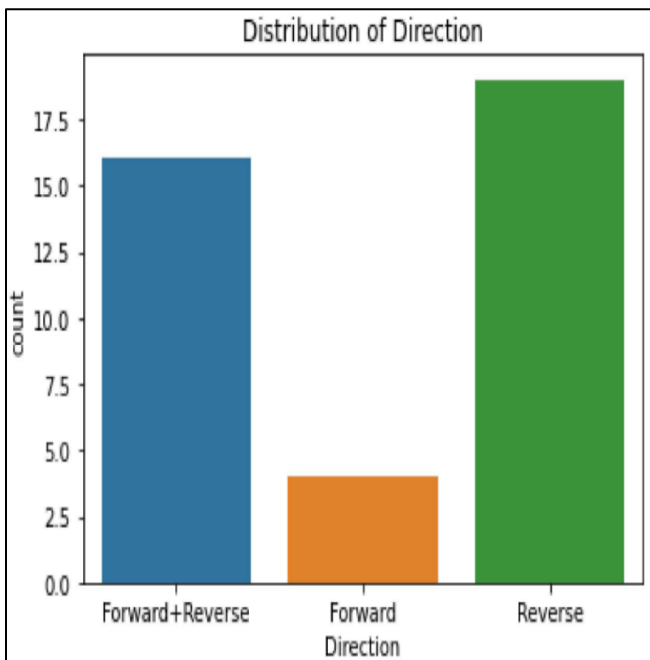
```
print(f'Accuracy: {accuracy}')
print('Confusion Matrix:')
print(conf_matrix)
```

```
Accuracy: 0.75
Confusion Matrix:
[[0 0 0 0 0]
 [0 3 0 0 0]
 [1 0 0 0 0]
 [0 0 0 2 0]
 [0 1 0 0 1]]
```

Fig 15 Accuracy of Model

- Above confusion matrix shows, distribution of true and false predictions. This confusion matrix will help organisation to decide the discount rates, which will be beneficial for both organisation and users.
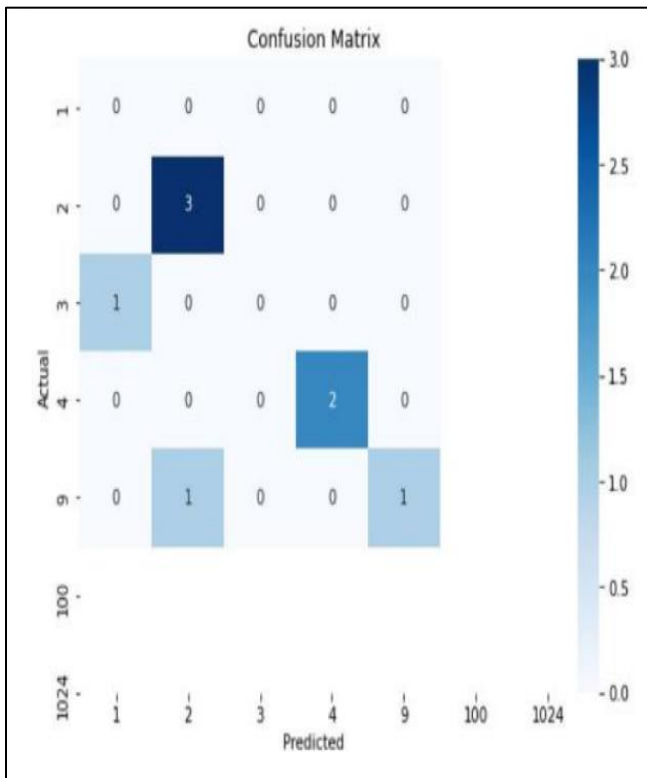
Fig 16 Confusion Matrix

V. CONCLUSION

In conclusion, the implementation of a streamlined billing process, enhanced accuracy, flexible billing options, seamless integration, scalable architecture, and compliance with security standards are paramount in ensuring customer satisfaction within the realm of billing systems for distributed computing environments. By addressing inefficiencies and challenges inherent in current billing systems, such as inaccuracies in resource usage measurement and lack of transparency, the proposed billing module offers a comprehensive solution. Its modular architecture enables seamless integration with existing Distributed Resource Management (DRM) systems, facilitating a smooth transition and minimizing disruption to operations. The provision of flexible billing options caters to diverse customer needs, while adherence to compliance and security standards ensures data integrity and user trust. Ultimately, the overarching goal is to optimize the billing process, improve customer satisfaction, and foster greater efficiency within distributed computing environments.

REFERENCES

[1]. Muzhir Shaban Al-Ani, Rabah Noory, and Dua'a Yaseen Al-Ani. Billing system design based on internet environment. *International Journal of Advanced Computer Science and Applications*, 3, 2012.

[2]. Muzhir Shaban Al-Ani, Rabah Noory, et al. Billing system design based on internet environment. *International Journal of Advanced Computer Science and Applications*, 3(9), 2012.

[3]. Belal Ayyoub, Bilal Zahran, Mahdi A. Nisirat, Farouq M. Al-Taweel, and Mohammad Al Khawaldah. A proposed cloud-based billers hub using secured e-payments system. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 2021.

[4]. Nijaz Bajgoric´, Lejla Turulja, and Amra Alagic´. Downtime and business continuity. In *Always-On Business: Aligning Enterprise Strategies and IT in the Digital Age*, pages 29–50. Springer, 2022.

[5]. Narendra Kumar Chahar, Krishan Pal Singh, and Muzzammil Hussain. Simplified micropayment mechanism to eliminate the risk of double payment in e-commerce. *2023 International Conference on Advances in Intelligent Computing and Applications (AICAPS)*, pages 1–6, 2023.

[6]. Jun Fang, Yanbo Han, and Zhuofeng Zhao. A hierarchical service community model for cross-organizational service management. In *2013 International Conference on Service Sciences (ICSS)*, pages 84–89. IEEE, 2013.

[7]. Zhou Fei, Deng Heping, Li Zhengyue, and Li Zhengfu. The analysis of e-commerce online payment status in china. In *2010 international conference on networking and digital society*, volume 2, pages 84–87. IEEE, 2010.

[8]. Georgios Gousios, Christos KK Loverdos, Panos Louridas, and Nec- tarios Koziris. Aquarium: An extensible billing platform for cloud infrastructures, 2012.

[9]. Jun Han, Kai Deng, and Xiaolei Deng. Privacy data synchronization method based on election in intranet. In *2022 3rd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, pages 498–501. IEEE, 2022.

[10]. Md Arif Hassan, Zarina Shukur, and Mohammad Kamrul Hasan. An efficient secure electronic payment system for e-commerce. *Comput.*, 9:66, 2020.

[11]. Yongqiang He, Yanrong Shi, and Aixiang He. Research on online payment mode of e-commerce. In *2010 IEEE International Conference on Software Engineering and Service Sciences*, pages 312–315. IEEE, 2010.

[12]. Yildiray Kabak and Asuman Dogac. A survey and analysis of elec- tronic business document standards. *ACM Computing Surveys (CSUR)*, 42(3):1–31, 2010.

[13]. Guiqin Li, Xuechao Deng, Maoheng Zhou, Qiuyu Zhu, Jian Lan, Hong Xia, and Peter Mitrouchev. Research on data monitoring system for intelligent ship. In *Advanced Manufacturing and Automation IX 9th*, pages 234–241. Springer, 2020.

[14]. Michael Netter and Gu¨nther Pernul. Integrating security patterns into the electronic invoicing process. In *2009 20th International Workshop on Database and Expert Systems Application*, pages 150–154. IEEE, 2009.

[15]. Kyaw Zay Oo. Design and implementation of electronic payment gateway for secure online payment system. *International Journal of Trend in Scientific Research and Development*, 2019.

[16]. Shristi Pant. A secure online payment system. 2011.

[17]. H T Ranjitha and Ms. Chethana. Advanced billing system for govern- ment departments. *International Journal of Advance Research, Ideas and Innovations in Technology*, 5:18–21, 2019.

[18]. Farida Setianingsih, Adhistya Erna Permanasari, and Warsun Najib. Management information system of the billing subsystem: A prototype design. *IJITEE (International Journal of Information Technology and Electrical Engineering)*, 2019.

[19]. KMS Soyjaudah, Pierre Clarel Catherine, and Irfaan Coonjah. Evalua- tion of udp tunnel for data replication in data centers and cloud environ- ment. In *2016 international conference on computing, communication and automation (ICCCA)*, pages 1217–1221. IEEE, 2016.

[20]. Davor Sˇpanic´, Denis Ristic´, and Boris Vrdoljak. An electronic invoicing system. In *Proceedings of the 11th International Conference on Telecommunications*, pages 149–156. IEEE, 2011.

[21]. Sharanya Pamadi Sridhar. E-commerce technology made easy. 2017.

[22]. Benzhi Sui, Qiuquan Li, Jun Wen, and Yuanzhe Yao. Billing system design of cloud services. In *2018 3rd International Conference on Control, Automation and Artificial Intelligence (CAAI 2018)*, pages 79–81. Atlantis Press, 2018.

[23]. Zvonimir Vanjak, Vedran Mornar, and Ivan Magdalenic. Deployment of e-invoice in croatia. In *ICSOFT (ISDM/ABF)*, pages 348–354, 2008.

[24]. Qifeng Yang, Zhengwei Cheng, and Ping Song. Research on online payment mode based on internet banking payment gateway. In *2007 In- ternational Conference on Convergence Information Technology (ICCIT 2007)*, pages 2043–2048. IEEE, 2007.