

SHODH SAMAGAM

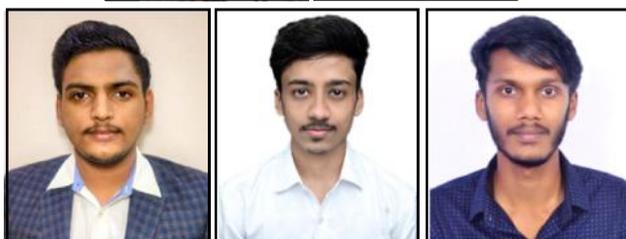
ISSN : 2581-6918 (Online), 2582-1792 (PRINT)



Invoice Management and Reconciliation System

**Prageet Bajpayi, CSE Department,
Akhlesh Soni, Somesh Awasthi, Ishan Singh, Akshat Bhanu,
B. Tech., 7th Semester, CSE Department,
Shri Shankaracharya Group of Institutions, Durg, Chhattisgarh, India**

ORIGINAL ARTICLE



Corresponding Authors

**Prageet Bajpayi, CSE Department,
Akhlesh Soni, Somesh Awasthi, Ishan Singh,
Akshat Bhanu,
B. Tech., 7th Semester, CSE Department,
Shri Shankaracharya Group of Institutions, Durg,
Chhattisgarh, India**

shodhsamagam1@gmail.com

Received on : 31/01/2022

Revised on : ----

Accepted on : 07/02/2022

Plagiarism : 07% on 31/01/2022



Plagiarism Checker X Originality Report

Similarity Found: 7%

Date: Monday, January 31, 2022

Statistics: 89 words Plagiarized / 1213 Total words

Remarks: Low Plagiarism Detected - Your Document needs Optional Improvement.

Invoice Management and Reconciliation System Akhlesh Soni,
soniakhlesh101@gmail.com B. Tech., 7th sem, CSE Department Abstract. Handling
invoices and reconciliation is a routine job which needs to be done but often delayed

ABSTRACT

Handling invoices and reconciliation is a routine job which needs to be done but often delayed because of its complex nature. Also getting all of this at same place is not possible. The existing system is manual handling where the vendor needed to save all the information offline and make all the reconciliation manually. The proposed system is more simple, error free and easily adaptable to any further changes. Invoice management system is built to handle the flow of invoices in an organization. The handling of this flow is sometimes hard which could eventually lead to mismanagement of records. Seamless Invoicing and Reconciliation System is a software that automatically manages invoices for rendered services and products. This software also provides a portal to the clients, so they can interact with their corresponding invoices to the vendor.

KEY WORDS

Reconciliation, Invoice Management, Transactions, Invoice.

INTRODUCTION

Invoice management and reconciliation system is a program to make the invoice management and reconciliation process easier for the companies and the customers. In this system we have two user roles, Admin and Customer. Admin has complete access of the system and full control over the actions. Admin can add and remove user. She/he can add, update and remove invoice. Admin have privilege to approve the reconciliation request from the customer. A Customer can read through invoices corresponding

to him/her. Customer can also either approve or raise an issue regarding drafted invoice. An issue is an object that is created when customer raises an issue over invoice. Customer can also make payment request or reconciliation request for the invoice. All the systems have implemented with React as frontend, Node.js as backend and PostgreSQL as Database.

Existing System

Handling invoices is much of routine work which needs to be done but often delayed mainly due to its tedious nature. Also getting access for various details of the same is also tedious. Earlier there existed manual system in which user needs to manage invoices respect to each of its customers manually. Also handling transaction and other issue was impossible. Although there are many systems in existence but they are still too complex for customer to understand. Our Designed system is way too simple for both the customer and the admin.

Proposed System

The proposed system is a portal where all the invoices are going to be uploaded and a flow will be created to handle all the actions that are going to be taken in order to complete the transaction. In this system admin will going to upload invoice as an entity. This invoice will going to have many fields like invoice reference and amount. After uploaded the invoice can be rejected by the customer and customer can also raise a issue regarding the invoice. If no issue is raised then customer can raise reconciliation request. This request contains fields like payment reference, amount and method. If admin approve this request then a transaction is created.

Json Web Token

JWT is a JavaScript library that helps in creating JWT token. A JWT Token is a time bound object which is created when we call sign method with payload and secret key as argument this key helps in verifying the token. This token expires in defined time. It also provides a standard way of authentication between two parties. It is secure, simple and easy to implement.

Implementation

Login

User module comprises of various data fields. Login requires only two fields which are email and password. Once user hits login button, the react frontend checks weather the information is entered or not, and is valid or not, once validated, an AJAX request is sent to NODE.JS server containing auth credential. On the server, received request is again goes for validation. After the validation, the email from the credential is used to find the user, if user is found then the password is also checked with a cryptic function whether the hash of the password is same or not, if the hash of the password is matched then a JWT object is created using user's email and a time object. This JWT object is called a Token. This token is then sent back to client with a success status. On client side, when the response from the server is received with a success status and JWT token, it get saved in LOCALSTORAGE as token and the user is redirected to dashboard page. In any of the above process if anything doesn't go according to the flow an message is displayed to the user error.

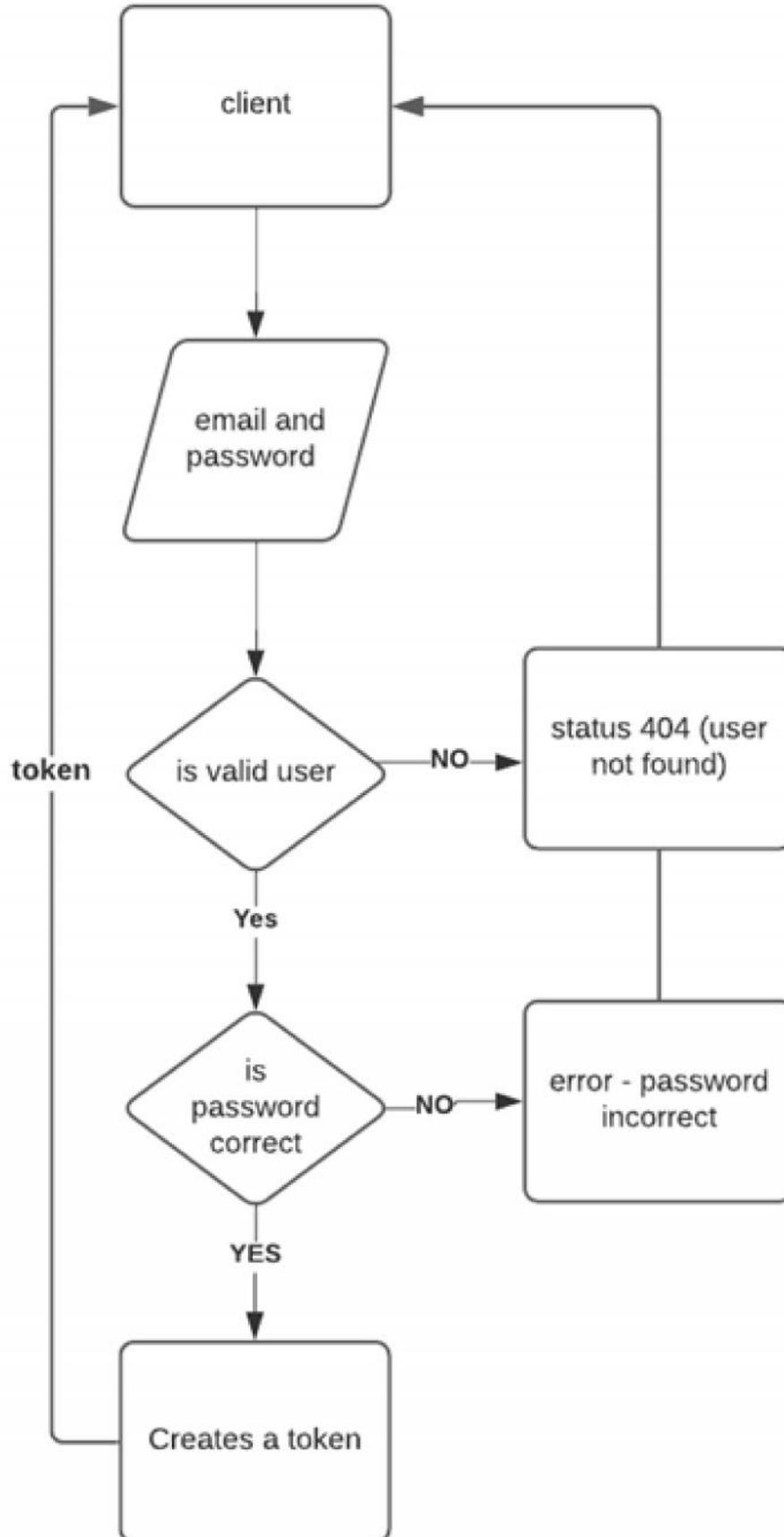


Figure 1: login process

Registration

An Admin can create both type of user, admin and customer. For adding a user, system has various fields which are required to be filled in order to add a user. Email is as primary field to identify rows uniquely. There are fields like USERTYPE, address, company and Phone Number.

After submitting the information is validated and An AJAX request is sent to the server containing user information. On the server, the information is again validated for possible error.

After all validation, user is added to the database with hash digest of the entered password as password.

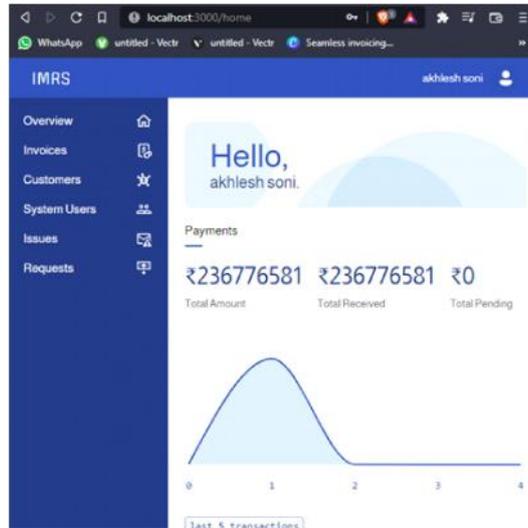


Figure 2: Dashboard page

Invoice Lifecycle

An invoice is a document that contains information about all the products and services provided by the vendor to a customer. In this system admin can add any number of invoices onto the system. For uploading invoice into the system, admin needs to fill some information and upload a file. The various fields are invoice id, date, due date, payment status, amount, paid amount and many others. They also need to upload an image file of the invoice they are referring to. When the invoice is added a success message is displayed to the Admin. Now when the customer sees the invoice they can either raise a issue or request for reconciliation.

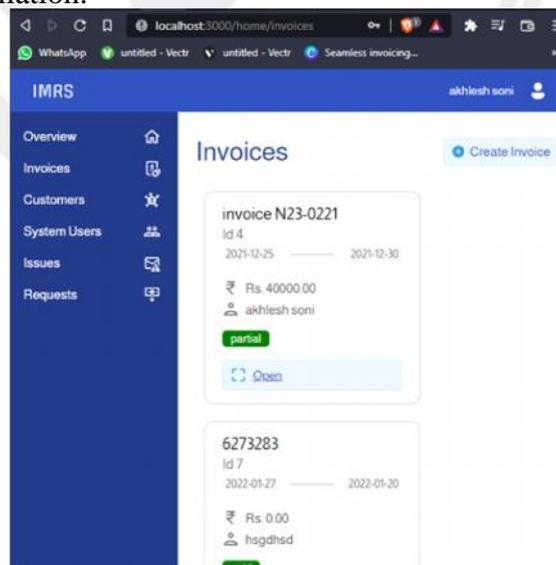


Figure 3: Invoice List Page

Issue Lifecycle

Issue is created when a customer raises it. After uploading invoice, if a customer finds any issue regarding invoice they can raise an issue. Issue contains various fields like id, invoice id, message and status. Admin only have right to read the issue, if the issue is resolved then customer can mark the issue as resolved to let admin know that they are satisfied. This System also saves all the resolved issues as history to avoid future conflict and ambiguity.

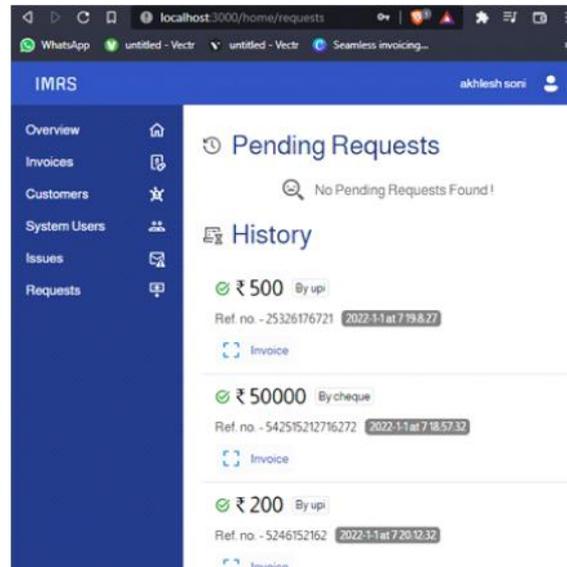


Figure 4: Request Page

Reconciliation Process

Reconciliation is the process of matching record and transaction for the clarification. In this system customer can raise a reconciliation request for a particular invoice. This reconciliation objects contains various fields like amount, payment method, payment reference and invoice id. After the request is created, if admin approves the request, a transaction is created, which is just a accepted reconciliation request.

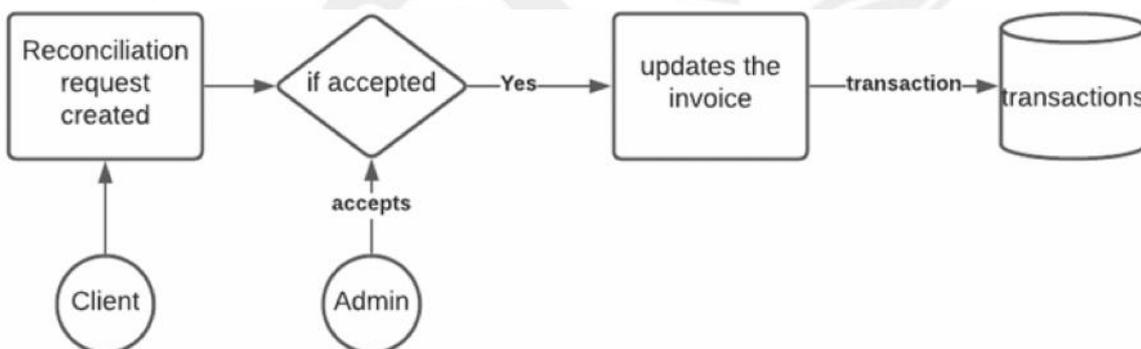


Figure 5: reconciliation flow

CONCLUSION

The system is being developed for the service. The easy handling and the simple UI have made the system accessible to all. The unique reconciliation system is also useful when making a request. The issue option has also made a big change on the whole vendor- customer relationship.

In future machine learning in scanning of invoice and wallet for customer to make advance payment will be added in the system.

REFERENCES

1. Adam Hayes, 2020, what is invoice, <https://www.investopedia.com/terms/i/invoice.asp> (accessed January 5, 2022).
2. Shelley Elmlad, 2021, what is account reconciliation, <https://www.thebalance.com/what-is-account-reconciliation-1293657> (accessed January 6, 2022).
3. Bezkoder, 2022, user authentication with jwt example, <https://www.bezkoder.com/react-express-authentication-jwt/> (accessed January 3, 2022).
4. Patrik Whatman, 2021, Invoice management for modern businesses, <https://blog.spendsk.com/en/invoice-management>, (accessd January 4, 2022).
5. Dienne Ledingham and Darrell Rigby, 2004, CRM done right, <https://hbr.org/2004/11/crm-done-right> (accessed January 3, 2022).
