Secure Complaint Management System against Women Harassment at Workplace Using Blockchain Technology

Original Scientific Paper

Md. Mijanur Rahman
Southeast University,
Assistant Professor, Department of Computer Science and Engineering
Banani, Dhaka, Bangladesh
mijanur.rahman@seu.edu.bd

Md. Moshiul Azam
Southeast University,
Student, Department of Computer Science and Engineering
Banani, Dhaka, Bangladesh
azammoshiul8@gmail.com

Faria Sanjida Chowdhury
Southeast University,
Student, Department of Computer Science and Engineering
Banani, Dhaka, Bangladesh
fariachy102@gmail.com

Abstract *– Since the Industrial era, women are playing a significant role in the workforce to move the world forward. Their increasing contribution in various fields has earned a fortune for the global economy. Despite that, women constantly face more obstacles than men in the workplace. When half of the population are mistreated because of gender inequality, the economy of any nation is supposed to collapse. One of the biggest barriers for women in their careers is workplace harassment. Workplace harassment may include physical, verbal or nonverbal harassment that not only have an adverse effect on a woman's career, mental health and physical health but also organizational reputation. A common way to make a complaint in most organizations is to fill up a complaint form, email or go directly to the competent authority and complain. But victims often hesitate to complain because their identity might get revealed or their documentary evidence might be tampered. As a result, most of the harassers get through very easily. To resolve this problem, this paper presents a blockchain-based anonymous, transparent and secure platform where women can easily complain against their harassers. To keep the platform secure and reliable, a two-level hierarchical model is introduced, where level-1 is the Human Resources (HR) and level-2 is the Higher Authority. In level-1, victims can anonymously complain to HR and in Level-2, victims can complain with their identity revealed to higher authority. This way, the proposed platform ensures women of a healthy work environment and provides all necessary support to stand up against injustice in the workplace.*

Keywords*: Blockchain, Anonymity, Hyperledger Fabric, Workplace Harassment, Women Harassment*

1. INTRODUCTION

Workplace harassment has been a serious issue for millions of working women across the world. It is defined as an offensive behaviour towards an employee by another to hurt them physically or mentally on purpose [1]. Workplace harassment includes many different forms of harassment such as insulting [2], bullying [3], teasing, mobbing [4], threatening, work abuse, physical abuse, sexual advance [5], etc. In a developed country like the USA, one out of three women claims to have been sexually harassed in the workplace [6]. There are also other cases of women falling victim to physical, verbal and non-verbal harassment. Such incidents cause negative effects on the victim's physical, emotional and occupational well-being [7].

To help working women in the workplace, there are some existing countermeasures against workplace harassment, such as, anti-harassment and anti-discrimination policy, monitoring system, report tracking system, and grievance procedures. Normally, a victim complains to Human Resource Personnel directly via a complaint form, email or hotline. But many victims hesitate to complain this way or have a fear of their identity being exposed. In some organizations, web-based applications and mobile applications are used to report such harassment. Some IT industries, for example, Speakfully [8] and #NotMe [9] offer services to employers and employees to deal with harassment in the workplace. Female employees can use these platforms for reporting and documentation at their convenience. But there is a possibility of data being altered or tampered. So, the security and reliability of these platforms are questionable.

Using Blockchain-based complaint systems in the workplace to overcome such limitations can be a remarkable solution for all of us. In 2008, Bitcoin, a peer-to-peer cryptocurrency based on blockchain technology, was first introduced by Satoshi [10]. Blockchain is now widely acknowledged in various fields because of its more secure, transparent, and tamper-proof ledger [11]. It also has prominent features like anonymity and autonomy. This distributed public ledger uses the Merkle tree and Hash function for its encryption and depends on consensus mechanisms such as Proof of Work (PoW), Proof of Stake (PoS), Proof of Concept (PoC), etc., as required. As the complaint management system for workplace harassment needs to be anonymous, reliable, secure, transparent and tamper-proof, the features of blockchain technology meet all its requirements.

There are two challenges in building a secure, reliable and tamper-proof complaint management system.

- The complaint may contain the victim's identity or other sensitive information that the victim wants to keep hidden. But, the authenticity and reliability of anonymous complaints cannot be guaranteed.
- The complaint needs to be transparent and tamper-proof as the authenticity and reliability of the complaint are the highest priorities. This is why traditional centralized systems are not considered to be secure and trusted.

Therefore, we proposed to build a decentralized complaint management system based on blockchain technology. Considering that blockchain has the trust of people for having decentralized, transparent, tamper-proof and trustless architecture, and it supports anonymity to assist a victim in filing their complaint without revealing their identity are two main reasons to utilize its properties in our system.

Our main research question is:

(1) How can blockchain help in supporting workplace harassment complaint procedures using its decentralized, secure, tamper-proof and trustless properties?

The objectives of our proposed system are as follows:

(1) Developing a blockchain-based complaint management system for employees and employers to deal with harassment in the workplace.

(2) Implementing two different levels to help victims complain either by keeping their identity hidden or revealing it.

2. RELATED WORKS

In 2020, Bárbara Aburachid Rocha proposed an Ethereum blockchain-based system for workplace harassment complaints and evidence tracking [12]. It documents the whole procedure while creating and tracking evidence of all the actions until the complaint procedure is resolved. The proposed architecture has two proof-of-concept. One is a fixed system that follows the guidelines of the Code of Practice Detailing Procedures for addressing Bullying in the Workplace from Ireland and the other is a flexible system that can be used in different procedures.

The Ministry of Women and Child Development, Government of India has an online workplace harassment complaint management system named SHe-Box for women where they can file complaints regarding sexual harassment at the workplace [13]. Any female employee regardless of any sector can use SHe-box. Victim has to provide her name, designation, contact number, email, identification number, accused's name, description, organization details, etc., to create a user id and file a complaint. She can also see the status of her complaint. However, SHe-box is a traditional centralized system, it is not transparent, tamper-proof and trustworthy.

Speakfully is a third party enabled app that helps employees report any incidents related to harassment and discrimination with documentation and support [8]. It offers different pricing solutions for individuals and organizations. Anonymous reporting with documents, case management, messaging with employees, pulse surveys, feedback, etc., are some of the features it provides. Here, victims have to enter their experiences in a document that can include text, image, audio, video, etc. They have to share personal data like name, email, contact number, address, company name, designation, IP address, etc., and report to their HR team. Such solution platforms are not very reliable as they work as a middleman.

3. SYSTEM BACKGROUND

In this section, the system background of our proposed system is explained.

A. Blockchain Technology

In 1991, two Bellcore researchers, Stuart Haber and W. Scott Stornetta established the first concept of blockchain technology [14]. Blockchain is a type of distributed ledger technology where every block stores data or transactions and system validators validate each block with a consensus mechanism. The first block of a chain is called "Genesis Block" [15]. To retain data, a block is recognized with hash and previous hash. Each block's previous hash is linked with the earlier block's hash. This is how blocks create chains (Fig. 1).

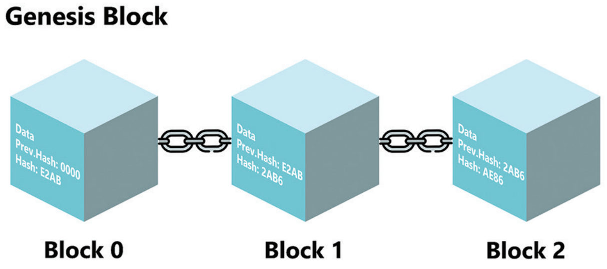


Fig. 1. Blockchain Technology

Blockchain is decentralized, immutable, secure, transparent and anonymous [11], [16].

Blockchain can be either permissioned blockchain or permissionless blockchain. Permissioned Blockchain is a closed ecosystem where only selected members can participate in the system with permission whereas permissionless blockchain is open for all. Anyone can participate in the system and validate any transaction [17].

B. Docker

Docker is a container similar to a virtual machine that allows developers to share containers and applications inside them among their peers [18]. It is an open-source project and works as a platform to build a server-client relationship. Docker images create containers [19] that can be considered as running instances of the base image. Alike an OS image, docker image executes code. It has multiple layers; each one is formed on top of the previous layer with commands while creating it. So, to share a code, developers have to share the image.

C. Hyperledger Fabric

Hyperledger Fabric is a popular project, introduced by IBM, with the intent of implementing a group of modular blockchain based applications within a single framework [20]. It is a partial permission based private blockchain, where every member has to get permission from the network, but the degree of permission varies for different users of different applications. Chaincode, aka, smart contract is used here in Docker containers [21]. It allows implementing applications in any programming language like Google Go, Node.js, Java etc.

D. CouchDB

CouchDB is a document-oriented database [22] that is supported by Hyperledger Fabric as a state database [23]. It is in JSON format that provides rich queries in opposition to chaincode making the queries more effective. Fabric stores the transaction data of its blocks in CouchDB as key and value pairs for user queries.

4. OUR PROPOSED FRAMEWORK

Our proposed system includes five types of entities-

- Victim
- Level-1
- Level-2
- Investigator
- Harasser/ Accused

Victim can complain on our proposed system in two ways.

1) Complaining anonymously to level-1: Victim wants the HR to warn their harasser.

2) Complaining with identity to level-2: In any serious case, the victim wants the higher authority to take immediate action against the offender.

A. Complaining Anonymously to Level-1

The following use case diagram (Fig. 2) shows how victim, level-1 and accused interact on our platform.

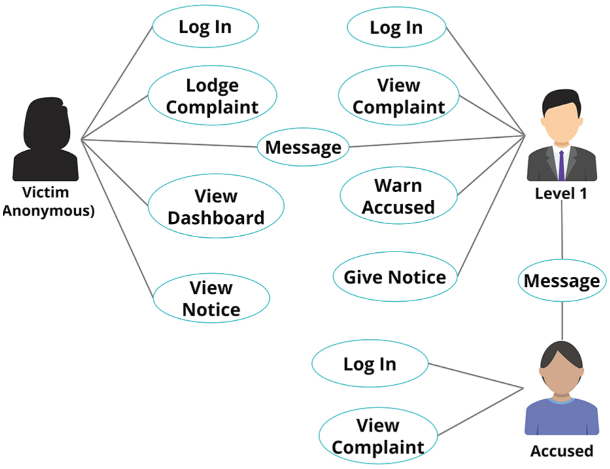


Fig. 2. Use Case Diagram for Level-1

Our proposed system for level-1 is divided into two systems.

a. Victim and Accused Interaction System

After logging in, the victim can anonymously lodge a complaint against her offender. She can choose what kind of complaint she wishes to file, usually minor cases of harassment. According to the complaint, Level-1 will give a warning to the accused. Both the level-1 and harasser will not know the identity of the victim.

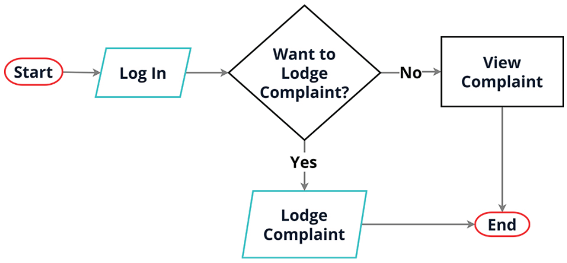


Fig. 3. Flow Chart of Victim and Accused Interaction System for Level-1

The following section describes Victim and Accused Interaction System (Fig. 3)

Lodge Complaint: If a victim wants to complaint, she has to provide complaint details, name of the accused and choose the complaint category.

View Complaint: Accused can view complaints filed against them.

b. Level-1 Interaction System

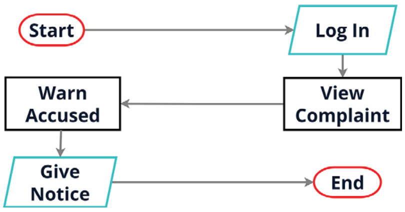


Fig. 4. Flow Chart of Level-1 Interaction System

The following section describes Level-1 Interaction System (Fig. 4)–

View Complaint: Level-1 can view complaints filed by victims and take action accordingly.

Warn Accused: If a complaint is filed against an employee, level-1 warns him.

Give Notice: Level-1 gives a notice about the action taken on the complaint.

B. Complaining with identity to Level-2

The following use case diagram (Fig. 5) shows how victim, level-2, accused and investigator interact on our platform.

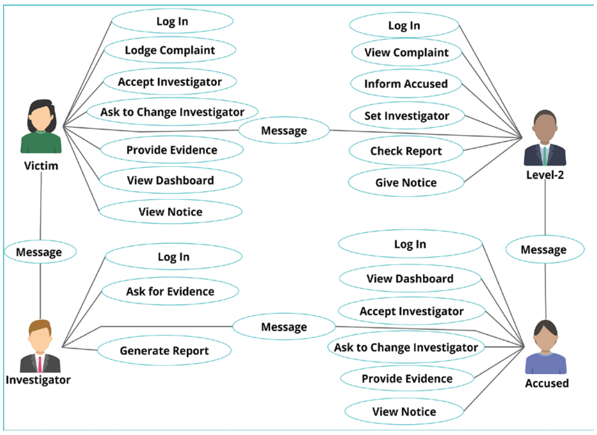


Fig. 5. Use Case Diagram for Level-2

Our proposed system for level-2 is categorized into three systems.

a. Victim and Accused Interaction System:

After logging in, the victim can lodge a complaint against her perpetrator disclosing her own identity. She can choose what kind of complaint she wishes to file. As for the accused, he will get a notification if he is alleged to have harassed a woman.

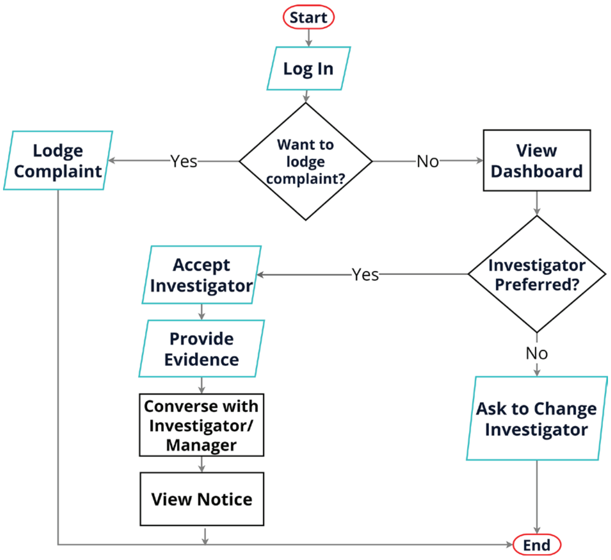


Fig. 6. Flow Chart of Victim and Accused Interaction System for Level-2

The following section describes Victim and Accused Interaction System (Fig. 6)-

Lodge Complaint: To lodge a complaint, a victim has to provide complaint details, name of the accused and choose the complaint category.

View Dashboard: Victim and accused can view their alleged complaint and get notified about the course of action.

Accept Investigator: Victim or accused will accept the investigator appointed by level-2 if they agree with the investigator to work on the matter.

Ask to Change Investigator: If any of them does not prefer the investigator, they can request level-2 to change the appointed investigator.

Provide Evidence: Victim or accused can provide their evidence to the investigator to conduct a thorough investigation.

Message: Victim and accused can message level-2 or investigator if needed.

View Notice: Victim, accused and other employees of the company will know what action has been taken on the complaint.

b. Level-2 Interaction System:

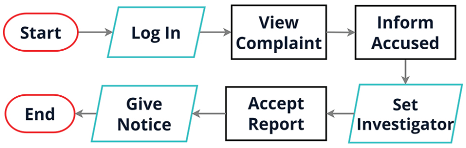


Fig. 7. Flow Chart of Level-2 Interaction System

The following section describes Level-2 Interaction System (Fig. 7).

View Complaint: Level-2 can view complaints filed by victims and take actions accordingly.

Inform Accused: If a complaint is filed against an employee, level-2 informs him.

Set Investigator: Level-2 appoints an investigator to scrutinize the evidence.

Message: Level-2 can message both the victim and the accused if needed.

Accept Report: Level-2 accepts the report generated by the investigator.

Give Notice: Level-2 gives a notice about the action taken on the complaint.

c. Investigator Interaction System

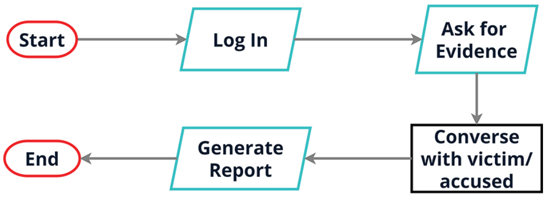


Fig. 8. Flow Chart of Investigator Interaction System

The following section describes Investigator Interaction System (Fig. 8).

Ask for Evidence: Investigator asks for evidence from the victim or the accused that may help him carry out the investigation.

Message: Investigator can message the victim and the accused if needed.

Generate Report: After completing the investigation, the investigator generates the investigation report to level-2.

There might be a situation in level-1 where the accused is not at fault and they want to take action against this false accusation. In this case, the accused can also complain to level-2.

While dealing with such a case, level-2 may need the previous complaint (Fig. 9)

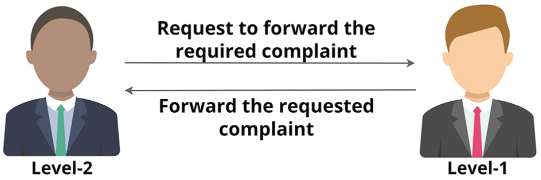

Request to forward the required complaint
Forward the requested complaint
Level-2
Level-1

Fig. 9. Flow Chart of the Interaction between Level-1 and Level-2

After the investigation is done, the authority either takes disciplinary action or legal action to support their victim at all costs. This way, an organization can confirm a safe workplace along with its positive reputation.

5. IMPLEMENTATION

Our proposed platform is a web application and uses Hyperledger Fabric blockchain to keep all records.

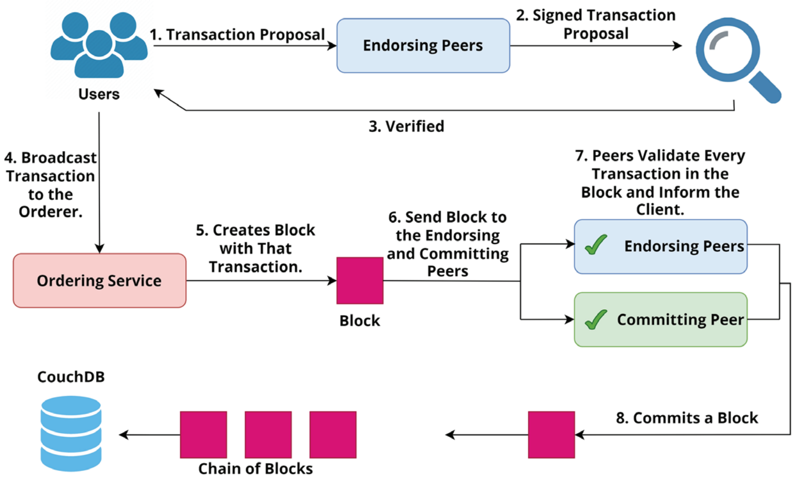

Users
1. Transaction Proposal
Endorsing Peers
2. Signed Transaction Proposal
3. Verified
4. Broadcast Transaction to the Orderer.
Ordering Service
5. Creates Block with That Transaction.
Block
6. Send Block to the Endorsing and Committing Peers
7. Peers Validate Every Transaction in the Block and Inform the Client.
Endorsing Peers
Committing Peer
8. Commits a Block
CouchDB
Chain of Blocks

Fig. 10. System Architecture

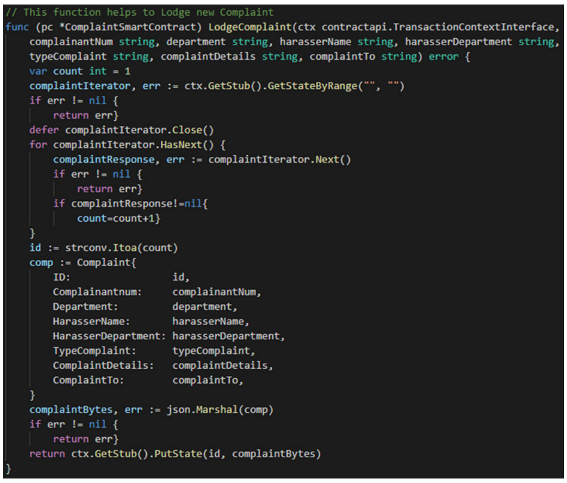
The following section describes the system architecture (Fig. 10)

Before interacting with the Hyperledger Fabric Network, the system verifies user (Victim/Level-1/Level-2/Accused/Investigator) identity using Membership Service Provider. After the user identification is confirmed, the network controls the user's access. When a user initiates a transaction, the transaction is broadcasted to the Endorsing Peers in form of a proposal. After receiving the proposal, Endorsing Peers execute the chaincode and return its consequences to the user. Then the user verifies the proposal response with the help of a consensus mechanism and broadcasts the transaction to the Ordering Service. Orderer creates a new block with that transaction and dispatches that block to the Endorsing and Committing Peers. Peers validate all the transactions in the block and notify the user too. Finally, committing nodes keep a copy of the block in their ledger.

In this proposed application, Next.js is used to design the front-end while the Fabric network acts as the back-end. To implement the Hyperledger Fabric Network, the chaincode is written in Golang which runs in a secured Docker Container. The Node SDK is used to set up communication between the front-end and back-end. It helps to query all the functions and properties from the Fabric Network and communicate with the CouchDB.

Some code snippets of our proposed platform are given below.

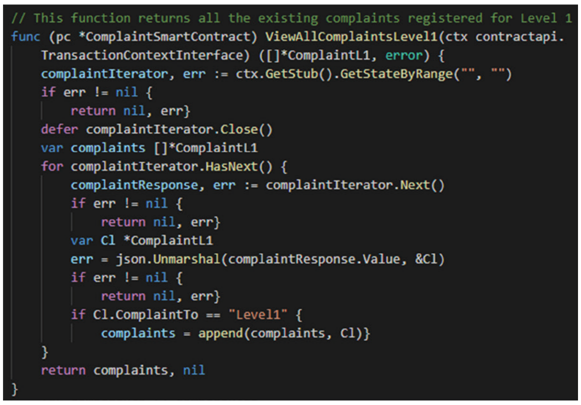
LodgeComplaint(): This function (Fig. 11) takes the complainant's ID, department, harasser's name, harasser's department, Type of complaint and complaint details from the complainant to lodge a complaint. Complainants can choose at which level they want to complain.



```go
// This function helps to Lodge new Complaint
func (pc *ComplaintSmartContract) LodgeComplaint(ctx contractapi.TransactionContextInterface,
    complainantNum string, department string, harasserName string, harasserDepartment string,
    typeComplaint string, complaintDetails string, complaintTo string) error {
    var count int = 1
    complaintIterator, err := ctx.GetStub().GetStateByRange("", "")
    if err != nil {
        return err}
    defer complaintIterator.Close()
    for complaintIterator.HasNext() {
        complaintResponse, err := complaintIterator.Next()
        if err != nil {
            return err}
        if complaintResponse!=nil{
            count=count+1}
    }
    id := strconv.Itoa(count)
    comp := Complaint{
        ID:                 id,
        Complainantnum:     complainantNum,
        Department:         department,
        HarasserName:       harasserName,
        HarasserDepartment: harasserDepartment,
        TypeComplaint:      typeComplaint,
        ComplaintDetails:   complaintDetails,
        ComplaintTo:        complaintTo,
    }
    complaintBytes, err := json.Marshal(comp)
    if err != nil {
        return err}
    return ctx.GetStub().PutState(id, complaintBytes)
}
```

Fig. 11. Function to Lodge a Complaint

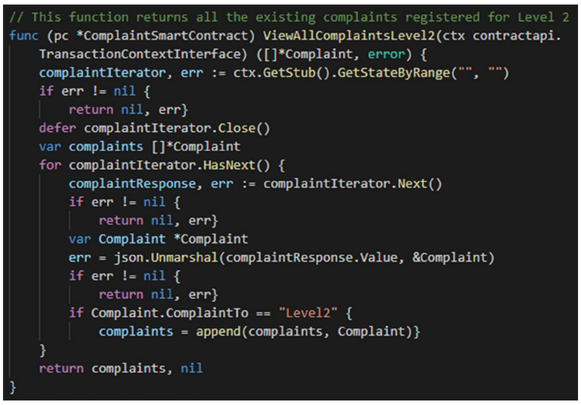
ViewAllComplaintsLevel1(): Level-1 uses this function (Fig. 12) to view all the complaints registered for them by the complainants. But here the complainant's ID is kept hidden, so here level-1 cannot see it and find out the victim's identity.



```go
// This function returns all the existing complaints registered for Level 1
func (pc *ComplaintSmartContract) ViewAllComplaintsLevel1(ctx contractapi.
    TransactionContextInterface) ([]*ComplaintL1, error) {
    complaintIterator, err := ctx.GetStub().GetStateByRange("", "")
    if err != nil {
        return nil, err}
    defer complaintIterator.Close()
    var complaints []*ComplaintL1
    for complaintIterator.HasNext() {
        complaintResponse, err := complaintIterator.Next()
        if err != nil {
            return nil, err}
        var Cl *ComplaintL1
        err = json.Unmarshal(complaintResponse.Value, &Cl)
        if err != nil {
            return nil, err}
        if Cl.ComplaintTo == "Level1" {
            complaints = append(complaints, Cl)}
    }
    return complaints, nil
}
```

Fig. 12. Function to View all Complaints by Level-1

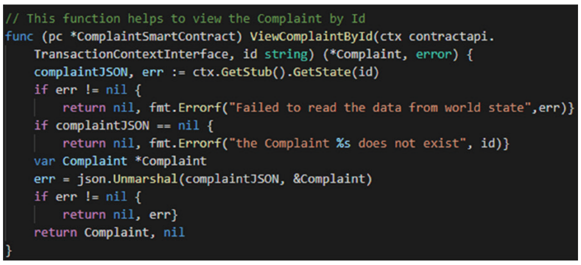
ViewAllComplaintsLevel2(): This function (Fig. 13) is used to view all the complaints registered for Level-2. Being the top authority, level-2 can see every information including the complainant's ID.



```go
// This function returns all the existing complaints registered for Level 2
func (pc *ComplaintSmartContract) ViewAllComplaintsLevel2(ctx contractapi.
    TransactionContextInterface) ([]*Complaint, error) {
    complaintIterator, err := ctx.GetStub().GetStateByRange("", "")
    if err != nil {
        return nil, err}
    defer complaintIterator.Close()
    var complaints []*Complaint
    for complaintIterator.HasNext() {
        complaintResponse, err := complaintIterator.Next()
        if err != nil {
            return nil, err}
        var Complaint *Complaint
        err = json.Unmarshal(complaintResponse.Value, &Complaint)
        if err != nil {
            return nil, err}
        if Complaint.ComplaintTo == "Level2" {
            complaints = append(complaints, Complaint)}
    }
    return complaints, nil
}
```

Fig. 13. Function to View all Complaints by Level-2

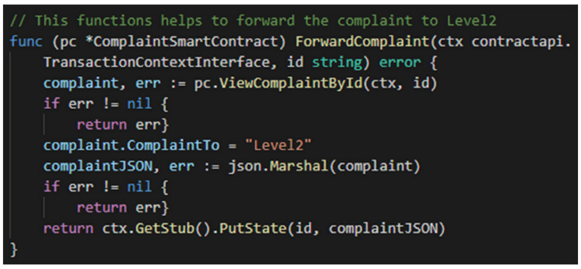
ViewComplaintByID(): Level-1 and level-2 use ViewComplaintByID (Fig.14) to view a complaint by its unique ID.



```go
// This function helps to view the Complaint by Id
func (pc *ComplaintSmartContract) ViewComplaintById(ctx contractapi.
    TransactionContextInterface, id string) (*Complaint, error) {
    complaintJSON, err := ctx.GetStub().GetState(id)
    if err != nil {
        return nil, fmt.Errorf("Failed to read the data from world state",err)}
    if complaintJSON == nil {
        return nil, fmt.Errorf("the Complaint %s does not exist", id)}
    var Complaint *Complaint
    err = json.Unmarshal(complaintJSON, &Complaint)
    if err != nil {
        return nil, err}
    return Complaint, nil
}
```

Fig. 14. Function to View a Complaint by ID

ForwardComplaint: This function (Fig.15) is used to forward a complaint. When level-2 needs a previous complaint to solve an issue and makes a request for it to level-1, level-1 uses this function to forward the complaint to level-2. Though level-1 does not know the identity of the complainant, the copy of the complaint discloses the identity only to level-2 after forwarding.



```go
// This functions helps to forward the complaint to Level2
func (pc *ComplaintSmartContract) ForwardComplaint(ctx contractapi.
    TransactionContextInterface, id string) error {
    complaint, err := pc.ViewComplaintById(ctx, id)
    if err != nil {
        return err}
    complaint.ComplaintTo = "Level2"
    complaintJSON, err := json.Marshal(complaint)
    if err != nil {
        return err}
    return ctx.GetStub().PutState(id, complaintJSON)
}
```

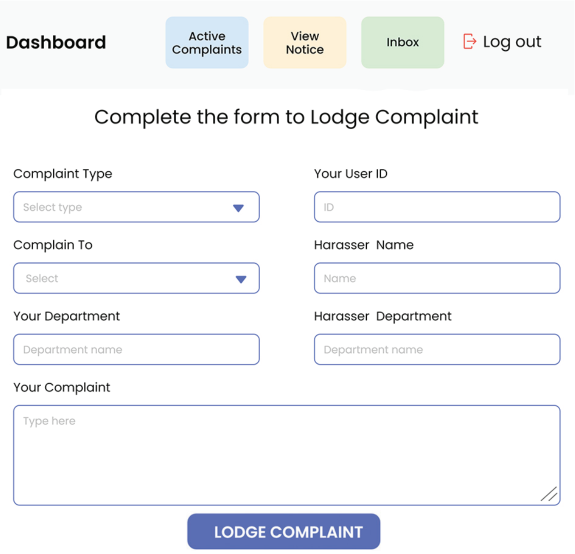
Fig. 15. Function to Forward a Complaint to Level2

6. RESULT

The proposed system has four modules– User, Level-1, Level-2, and Investigator. Level-2 is the highest level administrator while level-1 is the second level administrator. Level-1 and level-2 both control and manage system data. But level-1 has some restrictions here. Users are those who are employees of the company and are pre-registered in the system. They either file their complaints or get accused of their faults in the system.

On the other hand, an investigator is assigned to the system by level-2 to investigate a complaint.

When a user logs in, they can see the dashboard (Fig. 16). Here, they can file a complaint, check the statuses of their other active complaints or view any notice, such as warning for an accusation (only the accused can see it) and legal action or disciplinary action against other convicted employees.



Complete the form to Lodge Complaint

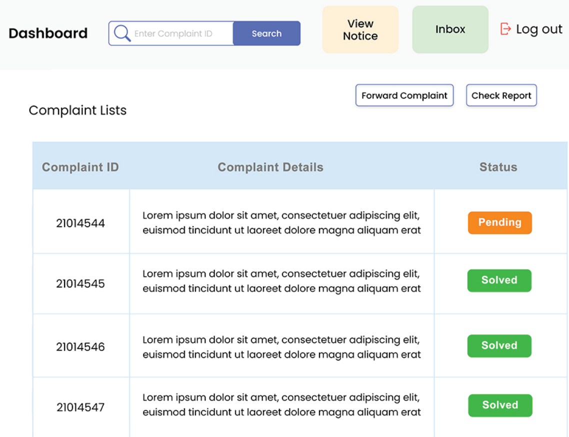
Complaint Type Select type	Your User ID ID
Complain To Select	Harasser Name Name
Your Department Department name	Harasser Department Department name

Your Complaint

Type here

Fig. 16. User Interface Dashboard

The following image (Fig. 17) is the dashboard for level-1 and level-2. They can view the complaint list, check reports generated by the investigator, view notices and use inbox to communicate with the users. They can also search a complaint by its unique id. If need be, they can forward a complaint to themselves or ask to be forwarded.

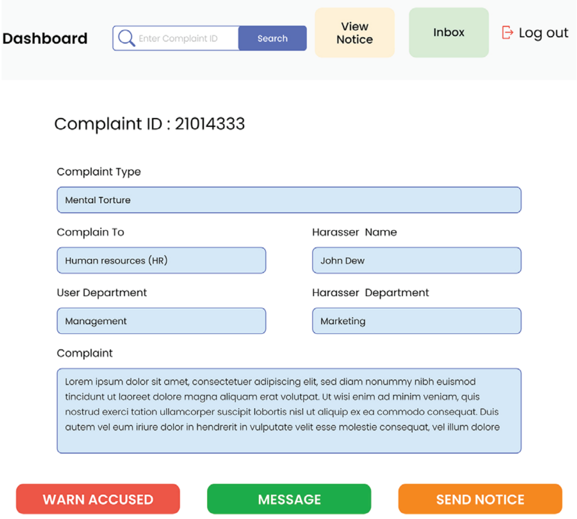


Complaint Lists

Complaint ID	Complaint Details	Status
21014544	Lorem ipsum dolor sit amet, consectetuer adipiscing elit, euismod tincidunt ut laoreet dolore magna aliquam erat	Pending
21014545	Lorem ipsum dolor sit amet, consectetuer adipiscing elit, euismod tincidunt ut laoreet dolore magna aliquam erat	Solved
21014546	Lorem ipsum dolor sit amet, consectetuer adipiscing elit, euismod tincidunt ut laoreet dolore magna aliquam erat	Solved
21014547	Lorem ipsum dolor sit amet, consectetuer adipiscing elit, euismod tincidunt ut laoreet dolore magna aliquam erat	Solved

Fig. 17. Level-1 and Level-2 Interface Dashboard

Level-1 and level-2 can access the details of any complaint from the complaint list. At Level-1, the identity of the victim is kept hidden (Fig. 18) while level-2, as the higher authority, can see the identity (Fig. 19). Both levels perform specific functions based on their administrative capacities.



Complaint ID : 21014333

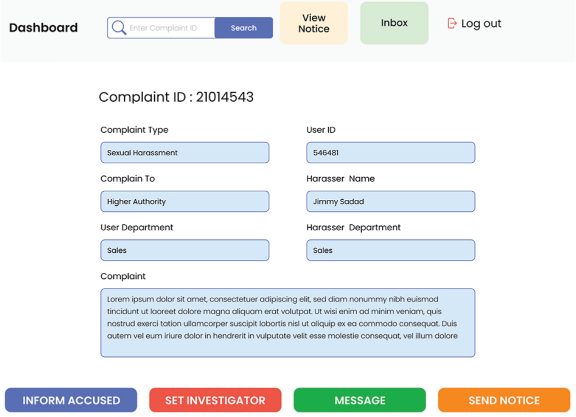
Complaint Type: Mental Torture

Complain To Human resources (HR)	Harasser Name John Dew
User Department Management	Harasser Department Marketing

Complaint

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore

Fig. 18. Complaint Details Interface for Level-1



Complaint ID : 21014543

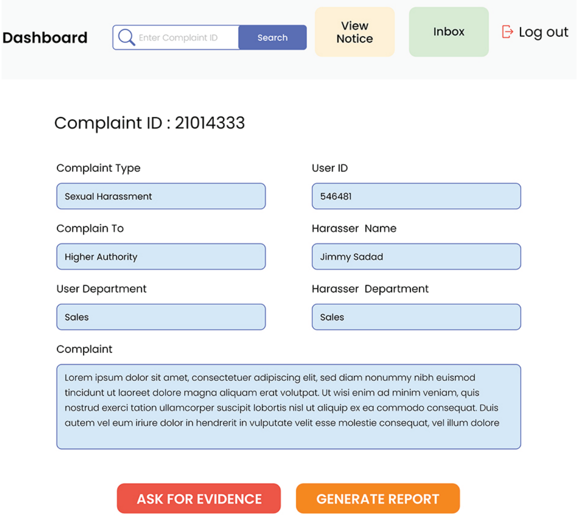
Complaint Type Sexual Harassment	User ID 546481
Complain To Higher Authority	Harasser Name Jimmy Sadad
User Department Sales	Harasser Department Sales

Complaint

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore

Fig. 19. Complaint Details Interface for Level-2

After being appointed by level-2, the investigator can view the details of the complaint on their dashboard (Fig. 20). They can ask for evidence from the victim and accused and may even message them if necessary. At the end of the investigation, they have to make a report to Level-2.



Complaint ID : 21014333

Complaint Type Sexual Harassment	User ID 546481
Complain To Higher Authority	Harasser Name Jimmy Sadad
User Department Sales	Harasser Department Sales

Complaint

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore

Fig. 20. Interface Dashboard for Investigator

7. CONCLUSION

This paper proposes a blockchain-based women harassment complaint system in the workplace for the documentation and management of complaints. The proposed system helps the victims to file their complaints in a trustless and tamper-proof environment while availing the prominent attributes like decentralization, anonymity, immutability, transparency, reliability and security of blockchain. It is composed of two hierarchical levels where level-1 ensures the victims' anonymity by letting them file complaints without revealing their identity and level-2 assists them in legitimately filing complaints with identity to the higher authority and documenting evidence. It not only makes the complaint management system more secure, productive and simple but also protects the victim from the next threat of the accused by preserving the victim's privacy. Furthermore, our system makes it easier for the HR and the higher authority to handle the complaints more efficiently. To cap it all, our proposed platform will ensure a safe working environment for women which will have a significant impact on the nation. Since very few papers have focused on this issue to solve with blockchain, our paper will contribute greatly to support the working women.

8. REFERENCES:

[1] K. Aquino, K. Lamertz, "A relational model of workplace victimization: Social roles and patterns of victimization in dyadic relationships", The Journal of applied psychology, Vol. 89, No. 6, 2004, pp. 1023–1034.

[2] S. Einarsen, "Harassment and bullying at work: A review of the Scandinavian approach", Aggression and Violent Behavior, Vol. 5, No. 4, 2000, pp. 379–401.

[3] J. E. Bartlett, M. E. Bartlett, "Workplace bullying: An integrative literature review", Advances in Developing Human Resources, Vol. 13, No. 1, 2011, pp. 69–84.

[4] S. B. Matthiesen, S. Einarsen, "Bullying in the workplace: definition, prevalence, antecedents and consequences", International Journal of Organization Theory & Behavior, Vol. 13, No. 2, 2010, pp. 202–248.

[5] A. Amin, M. S. Darrag, "Sexual Harassment in the Egyptian Workplace: A Literature Review and Research Agenda", Review of Management, Vol. 1, No. 4, pp. 25–38, 2011.

[6] B. L. Carvalho, M. E. Griffith, "More than One in Three Women Report Sexual Harassment In the Workplace", www.maristpoll.marist.edu (accessed: 2021)

[7] V. E. Sojo, R. E. Wood, A. E. Genat, "Harmful Workplace Experiences and Women's Occupational Well-Being: A Meta-Analysis", Psychology of Women Quarterly, Vol. 40, No. 1, 2016, pp. 10–40.

[8] "Speakfully | Workplace Reporting and Resource Platform." https://www.speakfully.com/ (accessed: 2021).

[9] "#NotMe." https://not-me.com/en/ (accessed: 2021).

[10] C. S. Wright, "Bitcoin: A Peer-to-Peer Electronic Cash System", SSRN Electronics Journal, 2019.

[11] K. Sultan, U. Ruhi, R. Lakhani, "Conceptualizing blockchains: Characteristics & applications", Proceedings of the 11th IADIS International Conference Information Systems, 2018, pp. 49–57.

[12] B. A. Rocha, "WORKPLACE HARASSMENT COMPLAINT EVIDENCE TRACKING SYSTEM USING BLOCKCHAIN", 2020,: https://harvest.usask.ca/handle/10388/13244 (accessed: 2021)

[13] "MINISTRY OF WOMEN & CHILD DEVELOPMENT." http://shebox.nic.in/ (accessed: 2021).

[14] S. Haber, W. S. Stornetta, "How to time-stamp a digital document", Journal of Cryptology volume, Vol. 3, No. 2, 1991, pp. 99–111.

[15] M. Crosby, Nachiappan, P. Pattanayak, S. Verma, V. Kalyanaraman, "BlockChain Technology: Beyond Bitcoin", 2016.

[16] H. F. Atlam, A. Alenezi, M. O. Alassafi, G. B. Wills, "Blockchain with Internet of Things: Benefits, challenges, and future directions", International Journal of Intelligent Systems and Applications, Vol. 10, No. 6, 2018, pp. 40–48.

[17] "Permissioned and Permissionless Blockchains: A Comprehensive Guide." https://www.blockchain-council.org/blockchain/permissioned-and-permissionless-blockchains-a-comprehensive-guide/ (accessed: 2021).

[18] C. Anderson, "Docker", IEEE Software, Vol. 32, No. 3, 2015, pp. 102–105.

[19] D. Bernstein, "Containers and cloud: From LXC to docker to kubernetes", IEEE Cloud Computing, Vol. 1, No. 3, 2014, pp. 81–84.

[20] "What is hyperledger fabric? | IBM." https://www.ibm.com/topics/hyperledger (accessed: 2021).

[21] M. Sethumadhavan, "On Blockchain Applications: Hyperledger Fabric And Ethereum", International Journal of Pure and Applied Mathematics, Vol. 118, No. 18, 2018, pp. 2965–2970.

[22] J. C. Anderson, J. Lehnardt, N. Slater, "CouchDB : the definitive guide", O'Reilly Media, Inc, 2010.

[23] E. Androulaki et al. "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains", Proceedings of the 13th EuroSys Conf. EuroSys, Porto, Portugal, 23-26 April 2018, pp. 1-15.