Blockchain Based Income Traceability System for Equitable Welfare Distribution

[1]Jithin S, [2]Muhammed Jinas T P, [3]Devathmaj A Kaliyathan, [4]Nevin R Pradeep, [5]Niranj C N

[1]Assistant Professor, [2]Student, [3]Student, [4]Student, [5]Student
[1]Department of Computer Science and Engineering,
[1]St. Thomas College of Engineering and Technology, Kannur, Kerala, India.

Abstract : Fair distribution of welfare benefits and accurate capture of income assessments for daily wage and informal sector workers in India are still a work in progress. Existing mechanisms such as income tax filings, government surveys like the Periodic Labour Force Survey (PLFS), socio-economic databases including the SECC, ration card records, and banking data provide fragmented representations of household income in India [8]. This challenge is compounded by the structure of the labour market, where around 80% of the workforce is engaged in the informal sector [9]. Consequently, income data is unreliable, prone to manipulation, or self-declared. As a result, many individuals claim to be Below Poverty Line (BPL) resulting in an exaggerated BPL count which in turn leads to the inefficient allocation of welfare schemes. This further deprives the truly deserving beneficiaries and increases social and economic inequity. In this context, we present an Income Traceability System based on Blockchain and AI/ML technologies. With this system, every digital wage disbursement gets recorded in the private Blockchain which ensures an immutable and verifiable income history. AI models are used to estimate income bands, dynamically classify individuals against welfare, BPL or Above Poverty Line (APL) eligibility. Also, the system provides a monitoring dashboard to the government for real-time policymaking without compromising on privacy, providing context-based income profiling, and automated welfare eligibility assessments.

IndexTerms **- Periodic Labour Force Survey (PLFS), SocioEconomic Caste Census (SECC), Below Poverty Line (BPL), Above Poverty Line (APL).**

I. INTRODUCTION

In India, the equitable distribution of government welfare relies heavily on income-based classifications like APL and BPL[10] to identify and uplift marginalized communities. However, the integrity of this allocation is severely compromised by the current methods of income data collection. Manual census-based identification methods used for welfare targeting in India have been shown to generate substantial inclusion and exclusion errors in identifying households eligible for welfare schemes [11]

The core of this inaccuracy lies in the informal economy. While formal sector wages are documented through regulated employment systems, incomes generated through informal economic activity are difficult to measure because these activities are often unregistered and operate outside formal institutional arrangements[12]. This systemic loophole has resulted in alarming rates of misclassification, as evidenced by recent census statistics[11]. When welfare eligibility and tax obligations are determined by income classifications, inaccuracies in reported economic status can undermine the effectiveness of these systems. Such distortions may strain government resources, complicate the development of targeted welfare policies, and reduce the efficiency of assistance intended for vulnerable populations

To address this critical gap, we propose Tracient, a system integrating blockchain technology with the Unified Payments Interface (UPI) to capture a verifiable trail of annual financial transactions. Tracient creates an immutable, transparent storage system that captures real-world economic activity rather than relying on self-reported survey data. This paper outlines the architecture of Tracient, illustrating how a decentralized approach can effectively replace tedious manual data collection and guarantee a more precise, equitable framework for welfare distribution.

A. Problem Statement

The real struggle in India isn't just coming up with welfare schemes, it's making sure the help actually finds the right people. Right now, categories like APL and BPL depend on how much someone earns in a year, but gathering that data is a mess. We still rely on manual surveys that are often full of mistakes or just plain outdated[11].

It's frustrating because, while we can track office workers' salaries easily, we have almost no way to verify what someone in the informal sector truly makes. This creates a huge loophole. Plenty of people know how to play the system to get benefits they don't actually need, while the families who are truly struggling get pushed to the back of the line. This income concealment leads to 'Inclusion Error' which helps them be exempted from taxes[13].

We realized that instead of chasing people for paperwork, we should be looking at how they actually get paid. By combining UPI with blockchain, we can create a record of transactions that is impossible to fake or delete. It gives us a crystal-clear, honest look at income without the need for door-to-door surveys. Our proposal is all about using this tech to cut through the red tape. We want to replace the slow, manual "old way" of doing things with a system that is fair, automatic, and actually reflects the reality of what people are earning. It's about making sure that when the government offers a hand up, it's reaching the person who needs it most.

II. LITERATURE SURVEY

Reviewing existing research was a vital step for us; it helped us see where the current technology stands and, more importantly, where it falls short for informal workers. While no single project addressed our specific goal of income traceability for welfare, we found four key papers that gave us the technical foundation we needed.

We focused on *"Blockchain-Integrated Digital Payment Systems in the Public Sector"* to see how this technology works in emerging markets. For the security aspect, we studied *"On the Integration of Artificial Intelligence and Blockchain Technology."* We also examined *"Blockchain-Based KYC and Access Verification"* to understand identity management, and *"An Explainable Federated Blockchain Framework"* for insights on maintaining data privacy while using AI.

These papers served as the building blocks for the security and transparency mechanisms used in our system.

A. Blockchain-Integrated Digital Payment Systems in the Public Sector: Innovation, Transparency, and Economic Efficiency in Emerging Markets [1].

This paper explores how blockchain can improve India's current government payment systems. Since UPI is already a massive success and used by almost everyone, the authors chose it as their base. They tried to build a hybrid model that uses blockchain to fix some of UPI's transparency gaps while also looking at the growing pains of such an implementation.The authors suggest a two-layer setup. The first layer is the standard UPI system we use every day because it is fast and scalable. The second layer is a blockchain that handles high-stakes government transactions where you absolutely need a record that cannot be changed or deleted.The paper's SWOT analysis gives a very honest look at the pros and cons. The big strengths are the speed of UPI combined with the unshakeable security of blockchain. On the other hand, the authors warn that connecting a decentralized blockchain to a centralized government system is technically difficult and expensive. This served as a great risk assessment for our own project.

B. On the Integration of Artificial Intelligence and Blockchain Technology: A Perspective About Security [2].

This paper explains how to take "centralized" AI and make it work on a "decentralized" blockchain. This is important because it allows the system to make decisions collectively rather than relying on one single server. The authors suggest a three-layer framework consisting of data generation at the top, an AI core in the middle, and a decentralized blockchain layer at the bottom. This setup makes the AI more ethical and secure, which gave us the roadmap for integrating our BPL/APL classification models into the blockchain.

C. An Explainable Federated Blockchain Framework with Privacy-Preserving AI Optimization for Securing Healthcare Data [3].

This paper introduces a framework called PPFBXAIO. Even though it is focused on healthcare, the problem it solves is exactly what we faced: how to analyze sensitive personal data without actually "seeing" it.The authors use "federated learning," which allows the AI to train on individual devices (such as a worker's phone) and only sends the "lessons learned" back to the main system. The blockchain then keeps a history of all the model updates.This was a major insight for us on how to keep worker income data private while still using AI to categorize them for welfare.

D. Blockchain-Based KYC and Access Verification for Financial Institutions [4].

This research looks at why traditional KYC (Know Your Customer) in banks is so slow and frustrating. Right now, it is mostly manual and centralized, which makes it an easy target for hackers. The paper proposes using a permissioned blockchain where only authorized parties can see the data. A clever trick used in the paper was storing actual sensitive documents in a secure separate location and only keeping the "digital fingerprints" (hashes) on the blockchain. This keeps the system fast while keeping personal data private. This helped us figure out how to handle worker identities in TRACIENT without exposing their private lives.

III. METHODOLOGY

This paper addresses the fundamental challenge of accurate income assessment for welfare classification in economies with significant informal sector participation. Traditional systems fail to capture the diverse, irregular, and undocumented income streams characteristic of informal work, leading to systematic misclassifications in BPL/APL determinations.

Target Population:

- **Formal Sector Workers:** Employees with documented wage records
- **Informal Sector Workers:** Daily wage laborers, street vendors, domestic workers, seasonal agricultural workers, construction workers, and other unorganized sector participants

A. AI Based Classification Model

1) **Multi-Criteria Decision Model:** The BPL/APL classification employs a hybrid approach combining SECC 2011 criteria [14] with multi-source income aggregation designed to handle both formal and informal sector income patterns.

Classification Function:

$$C(F) = \begin{cases} \text{BPL}, & \text{if } I_{total}(F) \leq T_{bpl} \vee D(F) \geq D_{threshold} \\ \text{APL}, & \text{if } I_{total}(F) > T_{bpl} \wedge D(F) < D_{threshold} \wedge E(F) = \emptyset \\ \text{pending}, & \text{if data insufficient or conflicting} \end{cases}$$

Where:

- C(F) = Classification of family F
- • Itotal(F) = Total annual income across formal and informal sources
- Tbpl = BPL threshold (200,000 for urban areas)
- D(F) = SECC deprivation score
- Dthreshold = Minimum deprivation indicators
- E(F) = Exclusion criteria violations

2) Unified Income Aggregation Model: Cross-Sector Income Calculation:

total(F) = Iformal(F) + Iinformal(F)

Where:

$$I_{formal}(F) = \sum_{i=1}^{n_f} S_i \times M_i \tag{3}$$

$$I_{informal}(F) = \sum_{j=1}^{n_i} W_j \times \bar{I}_{daily,j} \times D_j \times R_j \tag{4}$$

- Si = Monthly salary from formal employment i
- Mi = Months of formal employment
- Wj = Reliability weight for informal income source j
- Îdaily,j = Average daily income from informal source j
- Dj = Working days per year for source j
- Rj = Regularity factor (0-1) accounting for seasonal/irregular work

3) SECC Deprivation Scoring: Enhanced Deprivation Index:

$$D(F) = \sum_{k=1}^{7} \delta_k \times w_k + \alpha \times I_{volatility}(F)$$

Where:

- δk ∈ {0, 1} = Binary indicator for SECC deprivation criterion k
- wk = Weight assigned to criterion k
- α = Adjustment factor for income volatility
- Ivolatility (F) = Coefficient of variation in family income

4) Confidence Score Calculation: Sector-Adaptive Confidence:

Conf idence(C) = βformal × Cformal + βinformal × Cinformal

Where:

Cformal = Pmodel(class|X) × Documentationscore

Cinformal = Pmodel(class|X) × Consistencytemporal × Validationcross

B. Anomaly Detection Framework

1) Cross-Sector Income Pattern Analysis: Adaptive Outlier Detection:

$$Z_{adaptive} = \begin{cases} \frac{I_{current} - \mu_{sector}}{\sigma_{sector}}, & \text{if formal sector} \\ \frac{I_{current} - \mu_{seasonal}}{\sigma_{seasonal}}, & \text{if informal sector} \end{cases}$$

Misclassification Prevention:

$$Risk_{misclass} = \frac{|I_{reported} - I_{estimated}|}{I_{estimated}} \times Uncertainty_{factor}$$

2) Blockchain Data Integrity Verification: Multi-Source Validation:

$$Integrity_{total} = \prod_{s=1}^{S} Verify_{source}(s) \times Consistency_{temporal}(s)$$

Where S represents all income sources (formal and informal) for a family.

C. Hyperledger Fabric Network Architecture

1) Multi-Sector Network Configuration: Organizational Structure:

- Government Welfare Department (Policy Authority, MSP:WelfareMSP)
- Formal Sector Employers (Salary Records, MSP: FormalMSP)
- Informal Sector Aggregators (Daily Wage Records, MSP: InformalMSP)
- Verification Authority (Cross-Validation, MSP: AuditorMSP)
- Community Representatives (Local Validation, MSP: CommunityMSP)

Channel Architecture:

- Welfare-channel: BPL/APL classifications, eligibility records
- Formal-income-channel: Documented salary and wage transactions
- Informal-income-channel: Daily wage, piece-rate, and irregular income records
- Validation-channel: Cross-verification and audit trails

2) **Consensus and Endorsement: Multi-Party Endorsement Policy:**

E =”AND(’WelfareMSP.member’,OR(’FormalMSP.member’, ’InformalMSP.member’), ’AuditorMSP.member’)”

Algorithm 1 Cross-Sector Income Classification

function CLASSIFYWORKERFAMILY($familyData$)
  $I_f \leftarrow$ CALCFORMAL($familyData.fSrc$)
  $I_i \leftarrow$ CALCINFORMAL($familyData.iSrc$)
  $I_{tot} \leftarrow I_f + I_i$
  $secc \leftarrow$ EVALSECC($familyData.demo$)
  $vol \leftarrow$ CALCVOL($familyData.hist$)
  if $I_{tot} \leq$ BPL_THR **or** $secc \geq$ DEP_THR **then**
    $cf \leftarrow$ CALCCONF(I_f, I_i, vol)
    return {status: "BPL", conf: cf, bkdn: {f: I_f, i: I_i}}
  else
    $cf \leftarrow$ CALCCONF(I_f, I_i, vol)
    return {status: "APL", conf: cf, bkdn: {f: I_f, i: I_i}}
  end if
end function=0

D. Data Collection and Validation Framework

1) **Multi-Source Data Architecture: Formal Sector Data Sources:**

- Payroll management systems
- Tax filing records (Form 16, ITR)
- Provident fund contributions
- Bank salary credit records

Informal Sector Data Sources:

- Daily wage transaction records via mobile applications
- Peer validation systems
- Geolocation-based work verification
- Community representative attestations
- Market transaction records for vendors/traders

2) **Cross Validation Mechanism: Validation Metrics**

$$Accuracy_{overall} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision_{BPL} = \frac{TP_{BPL}}{TP_{BPL} + FP_{BPL}}$$

$$Recall_{APL} = \frac{TP_{APL}}{TP_{APL} + FN_{APL}}$$

$$Bias_{sector} = \frac{Classifications_{formal} - Classifications_{informal}}{Total_{classifications}}$$

Bias Prevention Measures:

1) Equal representation of formal and informal sector workers
2) Temporal income pattern analysis to account for seasonal variations
3) Community validation mechanisms for informal sector income claims
4) Regular algorithm auditing for sector-based classification bias

This methodology ensures equitable and accurate welfare classification across formal and informal sectors while preventing systematic misclassifications that have historically disadvantaged informal sector workers due to inadequate income documentation systems.

IV. PROPOSED SYSTEM

Our system, which we call Tracient, is a blockchain-based platform designed to track income and make welfare distribution more equitable. We built it to solve the massive lack of accountability in India's current system. By using verifiable data instead of just paperwork, we can accurately assess the income of the informal workforce. The system is built on four main layers:

A. Blockchain Layer

This is the base of the entire project. It creates a "trust infrastructure" where all income records are kept safe and cannot be altered once they are written.

a)Private Blockchain Network: We use Hyperledger Fabric [5] to run a permissioned blockchain network. This is not a public blockchain; instead, Fabric's Membership Service Provider (MSP) ensures that only authorized government bodies or employers can participate

b)Transaction Recording: Every bit of income data—whether it is a wage payment, a self-declaration, or a work record—gets hashed and stored. Because it is cryptographically secured, the history is traceable and completely resistant to anyone trying to "cook the books."

c)Smart Contract Validation: We wrote specific chaincode to handle the logic. These smart contracts do the heavy lifting: they initialize the ledger, manage UPI transaction records, and automatically apply the BPL/APL rules based on the state's specific laws. By putting the rules directly into the code, we make sure the auditing process is consistent and fair.

B. AI/ML Layer

The AI layer is there to spot people trying to cheat the system and to help categorize households accurately.

1) Machine Learning Models: We use a mix of models trained on historical data to keep the system sharp and accurate.

a)Transaction Anomaly Detection (XGBoost): To detect anomalous wage transaction patterns, we employ XGBoost [6], a scalable gradient boosting framework designed for high-performance machine learning on structured data. The model analyzes payment frequency and transaction amounts to identify irregular patterns that may indicate income underreporting or manipulation intended to retain BPL eligibility.

b)BPL/APL Classification (Random Forest): To classify households into BPL or APL categories, we employ a Random Forest classifier [**7**], an ensemble learning method well suited for handling heterogeneous socioeconomic features. The model can process multiple factors simultaneously, such as employment stability, income regularity, and reported household earnings, to produce a robust classification outcome..

c)Model Adaptability: The system isn't static. As new verified data hits the blockchain, the models retrain. This stops the AI from becoming outdated as the economy or income patterns change over time.

C. Dashboard and Decision Support Layer

This layer is for the people running the show. It's a clean interface for policymakers to see what's actually happening on the ground in real-time.

a)Administrative Dashboard: Government officials get a secure login to see anonymized, big-picture data. It's designed to be privacy-first, so they see trends and totals rather than prying into every single detail of a citizen's life.

b)Data Visualization and Analytics: Instead of messy spreadsheets, the dashboard uses charts and trend analysis. It makes it easy to see where the income gaps are and which areas are seeing the most growth or poverty.

D. User Interface Layer

This is how regular people and employers actually use Tracient. It has to be simple and secure.

a)User Registration and Role Management: The app lets you sign up as either a worker or an employer. Employers can log their workers' wages, while workers can keep their own household info updated.

b)Household and Family Information: We use ration card numbers to link individuals into family units. This is vital because welfare in India is usually calculated for the whole house, not just one person.

c)Welfare Eligibility Visualization: The best part for the user is the eligibility tracker. Based on the blockchain records and the AI's math, the app shows exactly which government schemes the user qualifies for. This helps solve the problem of people missing out on help simply because they didn't know it existed.

V. PROPOSED SYSTEM DESIGN

We designed this platform as a multi-layered web system that tracks income transparently and records wages securely. It is built to work for everyone, from formal office workers to those in the informal sector. We chose a microservices-based architecture because it keeps the system stable and makes it much easier to plug in different components as the project grows.

Users interact with the system through four specific web apps. Workers use their app to check wage history and see if they qualify for new benefits. Employers use a different interface to manage staff and log verified payments. Government officials have a dashboard to release new policies and spot income anomalies, while the system administrator keeps an eye on the overall health of the blockchain and the servers.

To keep things secure, every single request goes through a central API Gateway. This acts like a digital security guard, handling everything from login authentication to rate limiting. This setup ensures the backend stays protected even if the system gets a sudden surge of traffic. The ”brain” of the platform is the Application Service. It coordinates everything—validating requests and making sure the blockchain, policy rules, and database layers are all talking to each other correctly.

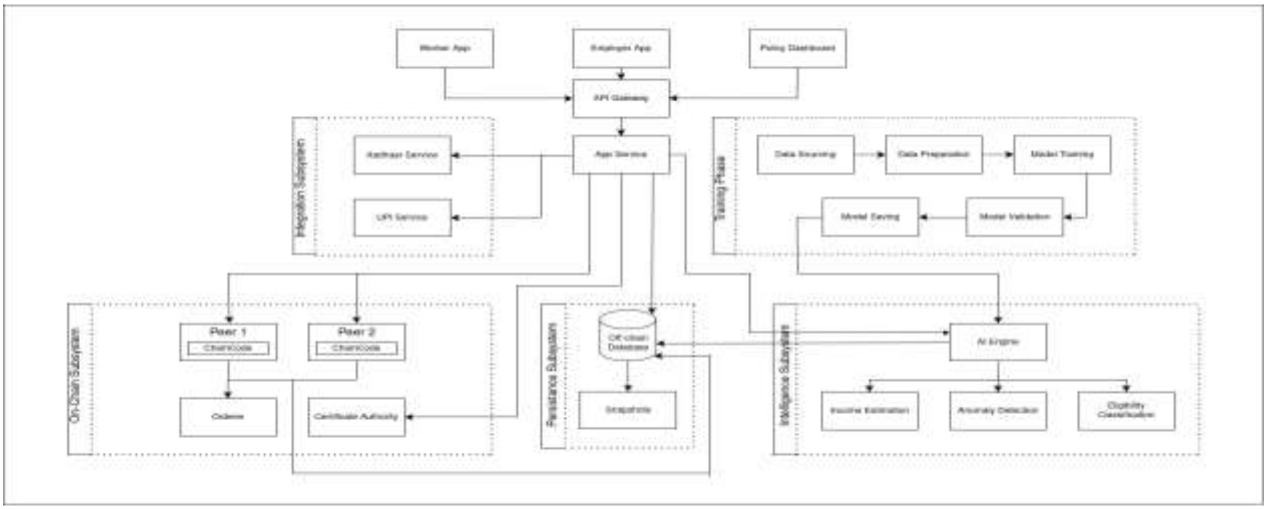


Figure 1. Architecture Diagram

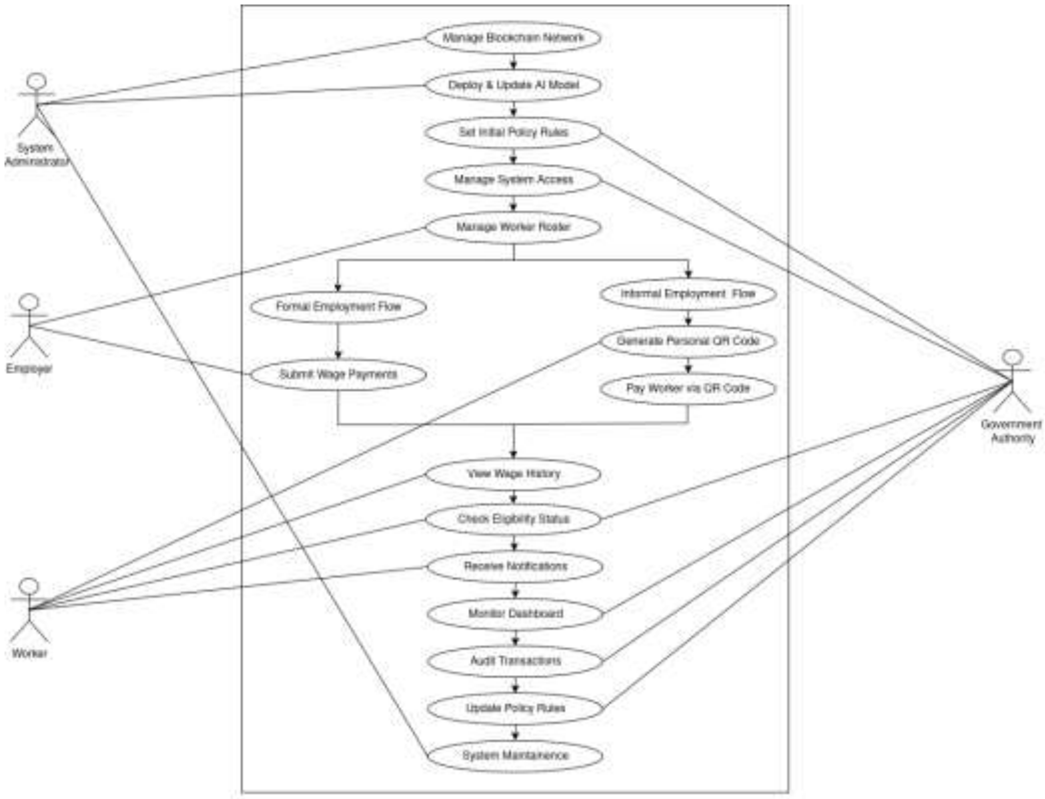


Figure 2. Use Case Diagram

Instead of working in a strict ”Step 1, Step 2” fashion, the system is a living environment where payments are processed and policies are updated at the same time. We have broken down how it functions into a few core workflows.

A. Policy and System Configuration

We built a Policy Administration Dashboard so that rules don't have to be hard-coded. This allows the government to add or remove welfare schemes on the fly without having to restart the entire system.

a)Policy Definition: Officials use this interface to set the actual numbers—like income cutoffs for BPL status or specific eligibility rules. Once updated, these rules immediately start applying to every new transaction that comes through.

b)System Initialization and Updates: The admin side handles the heavy lifting of managing the blockchain parameters and smart contracts. If there's a sudden spike in users, the administrator manages the load balancing to keep the site from slowing down.

B. Transaction Processing and Verification

This is the engine room of the platform where wages are actually submitted and locked into the record.

a)Wage Submission and Verification: When an employer sends a wage payment, the system doesn't just record it blindly. It first checks if the data is formatted correctly and follows current policy rules.

b)Blockchain Transaction Recording: Once a payment is validated, it's written to our Hyperledger Fabric network. Smart contracts enforce the business logic, while the ”orderer” makes sure every transaction is sequenced properly. We also use a Certificate Authority to ensure that only verified users can actually write to the ledger.

C. Eligibility Evaluation and Monitoring

The system never stops checking who qualifies for what. Our AI/ML model is constantly looking at the income data to update BPL and APL classifications.

a)Eligibility Assessment: As soon as new wage data is logged, the system re-runs its checks. This means a worker's eligibility status is always up-to-date, reflecting their real financial situation rather than a report from three years ago.

b)Monitoring and Auditing: The government gets a bird's-eye view of everything. They can look at anonymized trends to see how poverty levels are shifting. Because it's on a blockchain, they can also perform audits that are fully transparent and impossible to manipulate.

D. Integration and Output Delivery

No system works in a vacuum, so we designed this to hook into third-party tools like digital ID verification (like Aadhaar) and electronic payment gateways.

a)External Service Integration: No system works in a vacuum, so we designed this to hook into third-party tools like digital ID verification (like Aadhaar) and electronic payment gateways.

b)Persistence and Output: We use a persistence layer to store snapshots of the data. This makes the app feel fast because we don't have to query the entire blockchain for every single page load. All this data is then fed back to the users through their specific dashboards, making the whole process feel seamless and secure.

VI. RESULTS

In this section, we go over the actual data and performance results from our implementation of TRACIENT. These findings come directly from real system logs, our blockchain test environment, and the outputs of our AI classification models. The figures throughout this section illustrate how the platform looks and functions in a live setting.

A. System Deployment and Authentication Results

We successfully stood up the TRACIENT platform as a web-based system that talks to a private Hyperledger Fabric network and an AI analysis engine. One of our first tests was ensuring that the role-based access actually worked.

As you can see in Fig. 3, the login screen manages access for different groups—workers, employers, and government officials. When a user logs in, the system correctly identifies their role and pushes them to the right dashboard, confirming that our session management and security layers are solid.

B. Blockchain and Smart Contract Performance

We used a dedicated testing interface to verify that our Hyperledger Fabric network was handling requests properly. Fig. 4 shows the system connected to the blockchain channel, where the chaincode is live and ready for queries.

Our tests showed that query functions, like pulling up a worker's wage history, worked instantly. More importantly, we confirmed that only authorized users could actually ”submit” or change data on the ledger. This proves that our endorsement policies and smart contracts are doing their job of keeping the data tamper-proof.

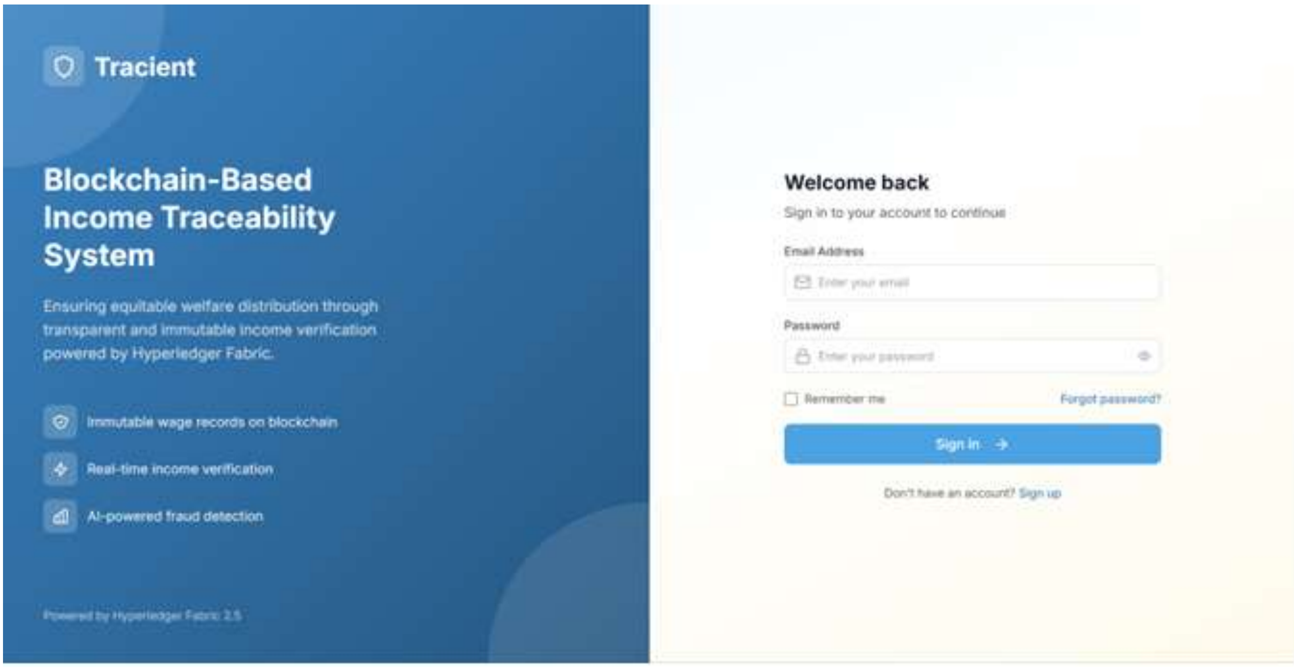


Figure 3. The TRACIENT login screen showing how we handle different user role.

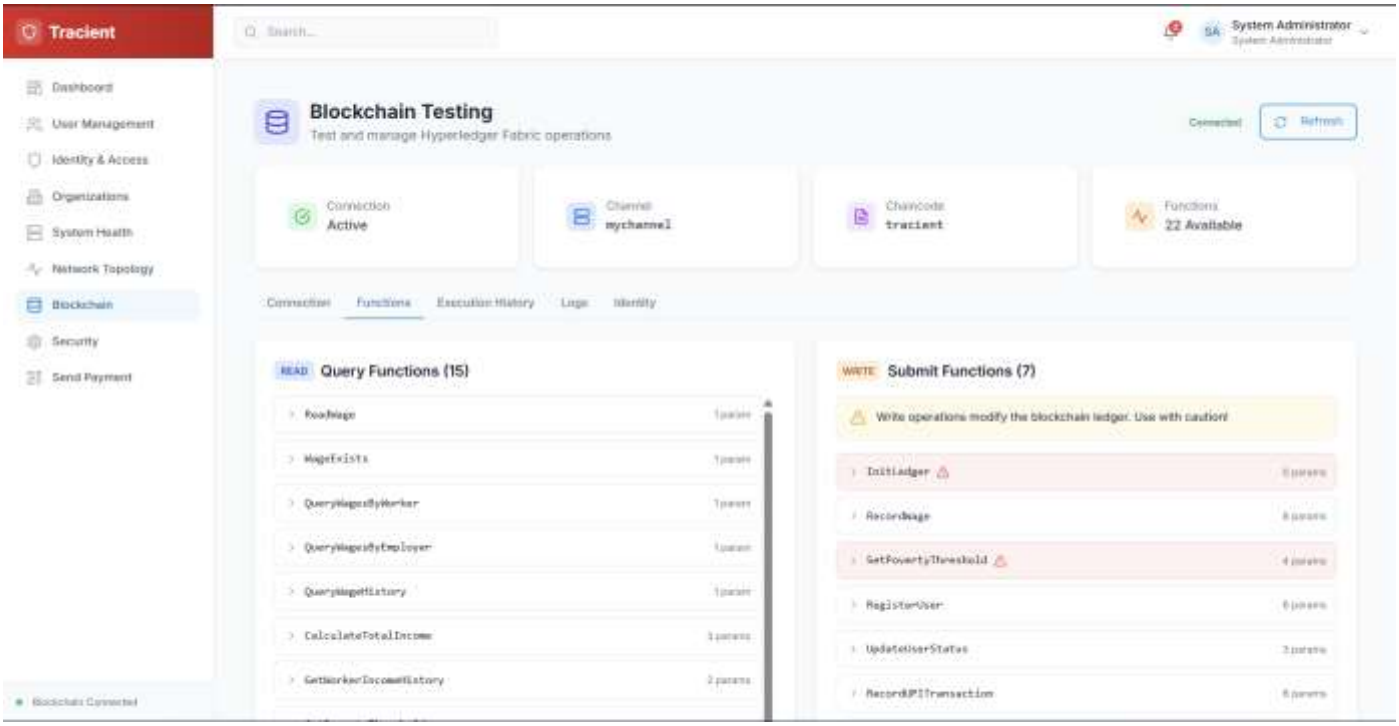


Figure 4. Testing the Fabric network to ensure smart contracts and queries are running correctly.

C. Wage Recording and QR Payment Results

A big part of our system is making payments easy and traceable. We built a module where workers can get paid via a QR code linked to their bank.

In Fig. 5, the interface shows how the QR is generated while keeping sensitive bank details masked for privacy. Once the payment goes through, the system immediately writes an unchangeable record of that wage to the blockchain.

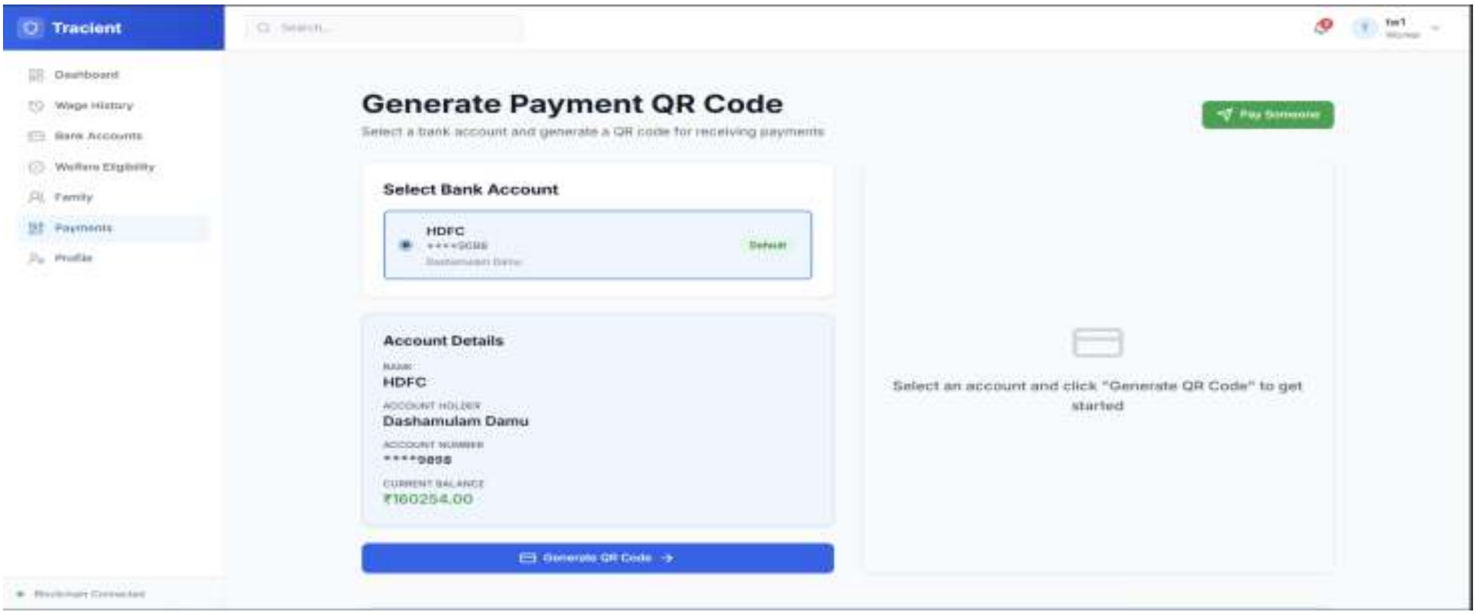


Figure 5. Our QR payment interface, which links digital payments directly to the blockchain ledger.

D. AI Family Profiling and Welfare Outcomes

The AI module is responsible for the actual ”heavy lifting” of determining who gets welfare. As shown in Fig. 6, the system builds a profile for each household.

It looks at family size, literacy, and job types—not just a single income number. By combining this demographic data with the verified income history from the blockchain, the AI can categorize families into BPL or APL status much more accurately than a manual survey ever could.

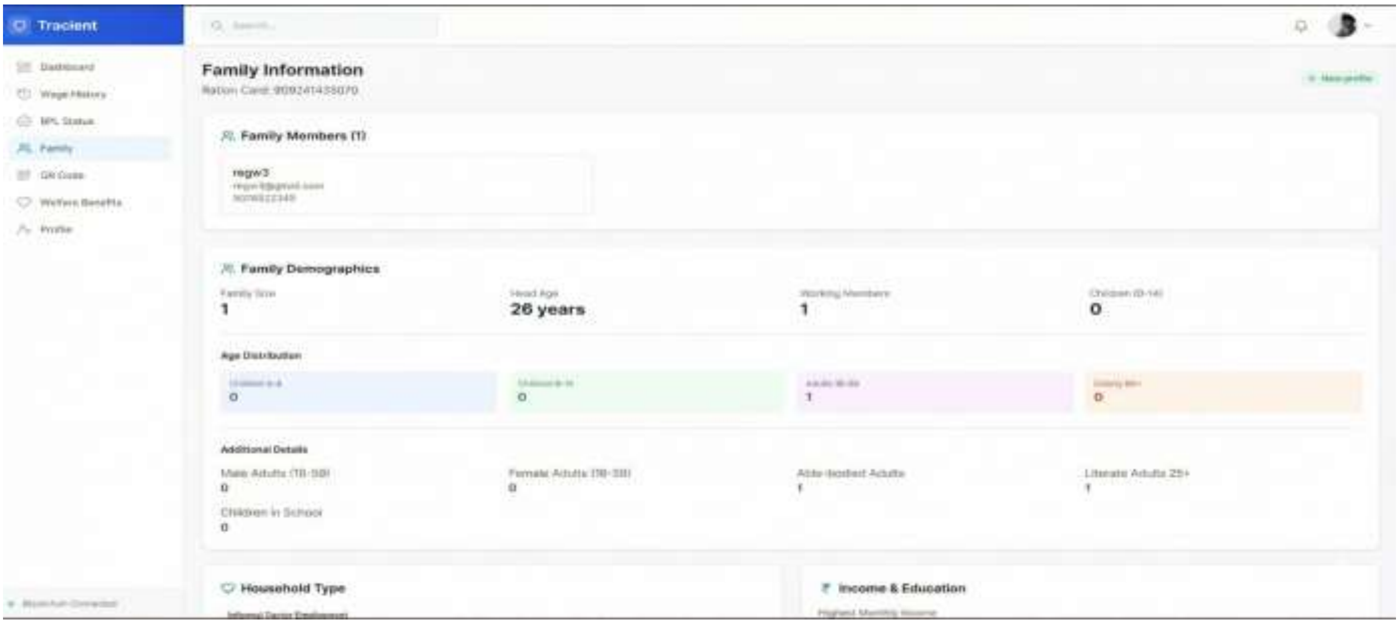


Figure 6. The profiling screen where the AI evaluates household data for welfare eligibility.

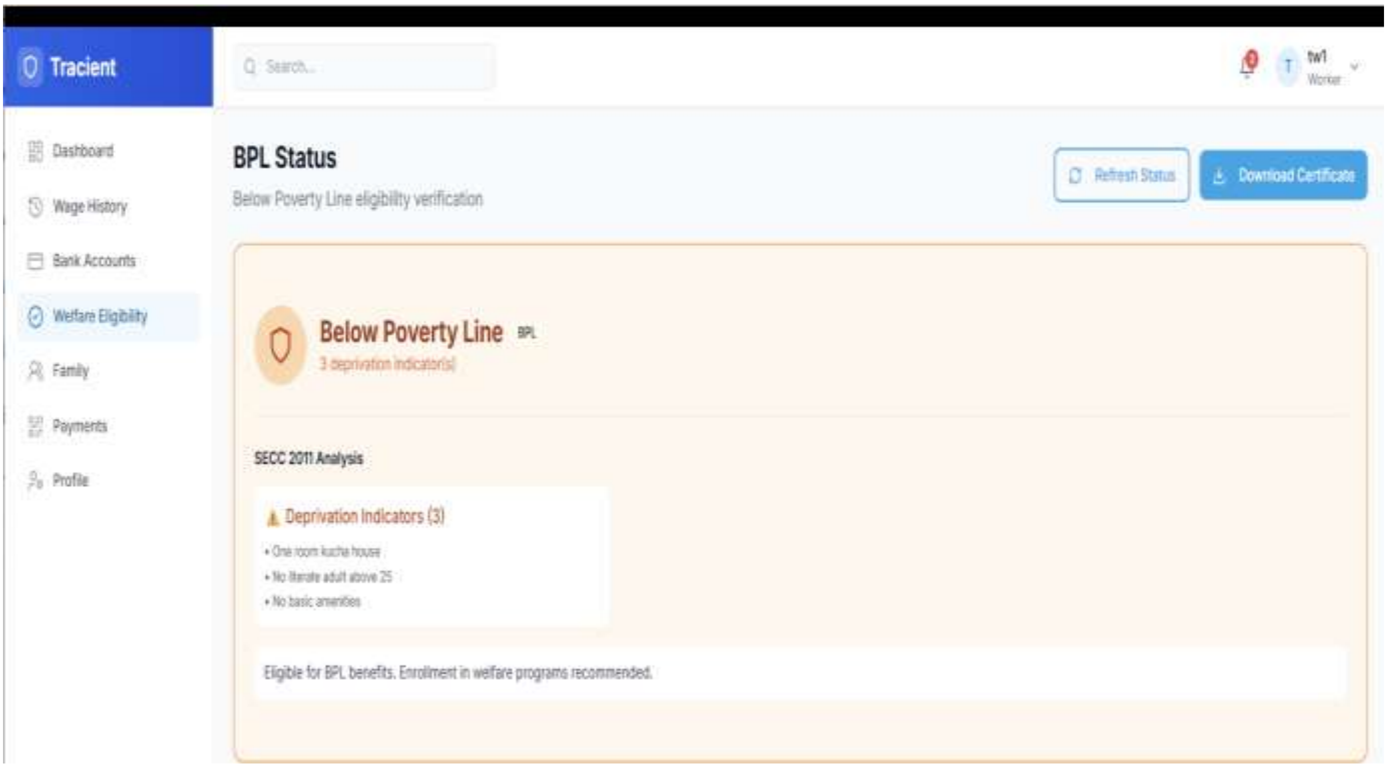


Figure 7. Category Classification by APL/BPL Classification model.

E. Anomaly Detection Model

For evaluating the working of our Anomaly Detection model, we injected test data containing anomalous transactions among a large number of normal transactions. The model accurately identified them and pointed out the accounts to which these transactions were made as shown in Fig. 8.

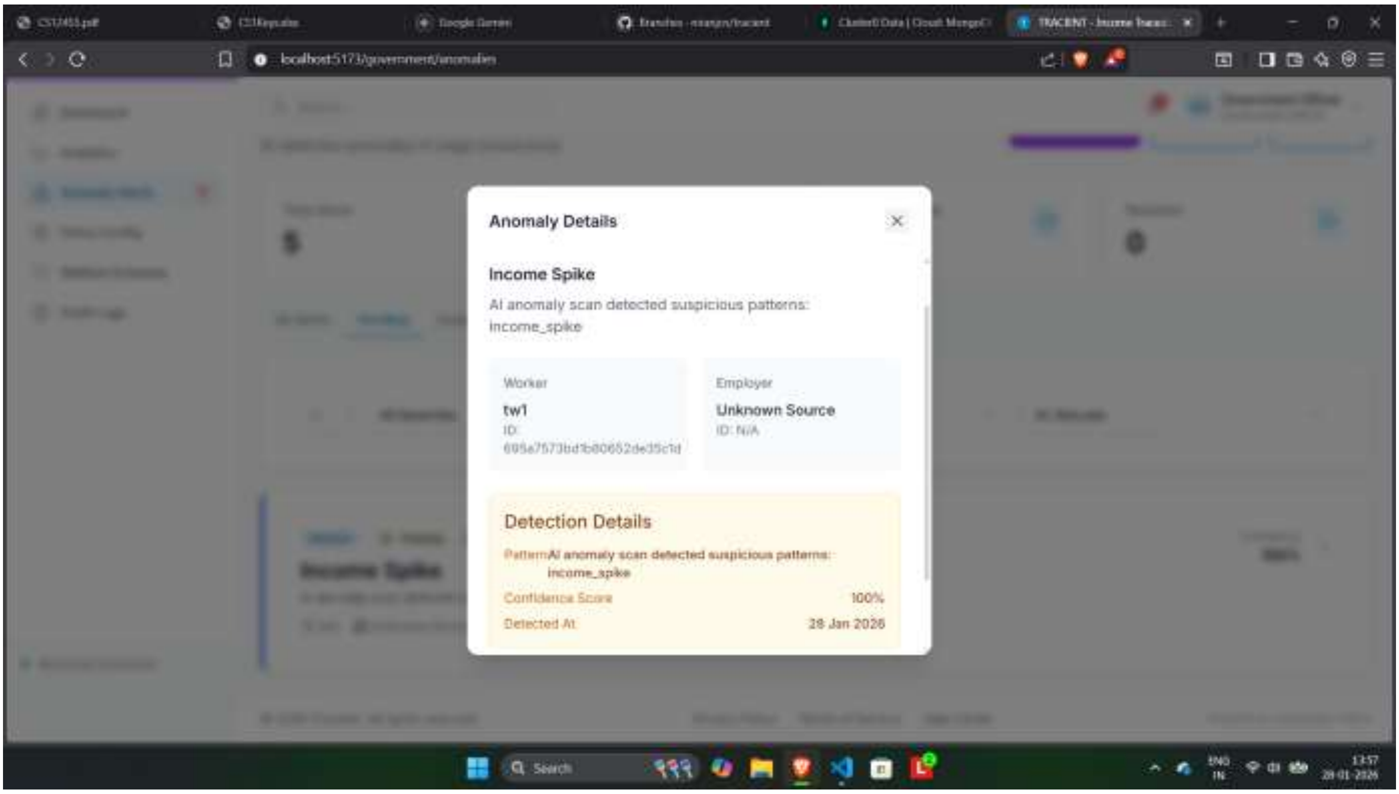


Figure 8. Income Spike Identified as Anomaly

F. Policy Configuration and Re-classification

To test the Policy Configuration and Re-classification module, we considered a set of user accounts belonging to APL or BPL categories that had a single cause for classification. After imposing various policies with different priority levels, we were able to observe the users being reclassified into appropriate categories (Fig. 9).

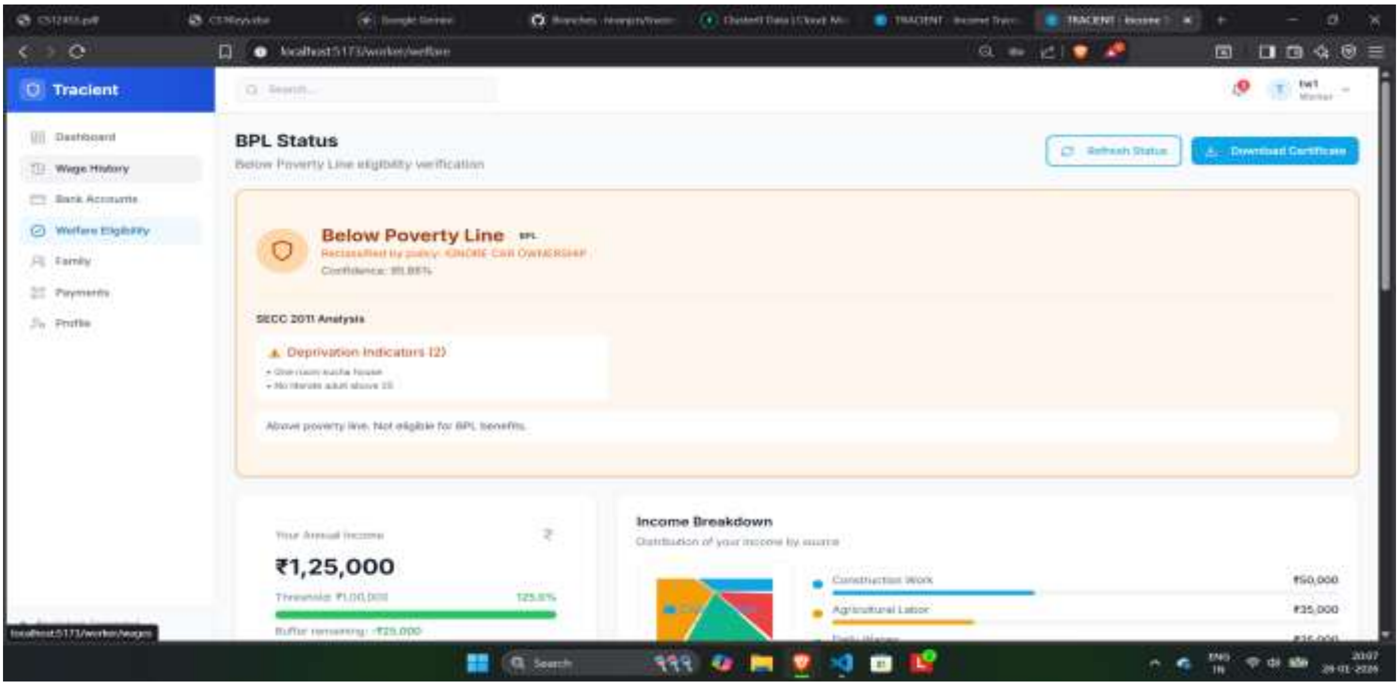
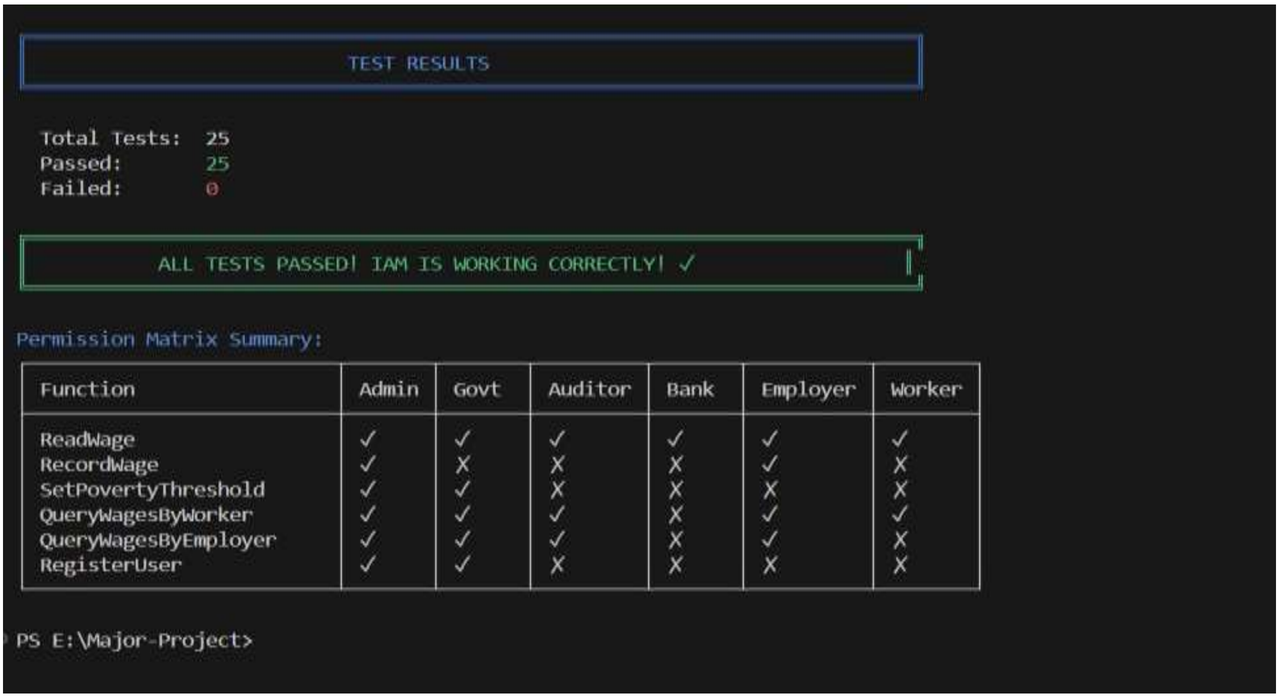


Figure 9. Re-classification based on Policy Change

G. Smart Contract Reliability and Access Control

We put our smart contracts through a rigorous battery of tests to make sure there were no bugs in the logic. Fig. 10 shows that every single unit test passed without a hitch.

We specifically checked the "permission matrix" to ensure a worker couldn't change poverty thresholds or an employer couldn't view someone else's family data. The results confirm that the system follows our security rules perfectly.



```
TEST RESULTS

Total Tests:  25
Passed:       25
Failed:       0

ALL TESTS PASSED! IAM IS WORKING CORRECTLY! ✓

Permission Matrix Summary:
```

Function	Admin	Govt	Auditor	Bank	Employer	Worker
ReadWage	✓	✓	✓	✓	✓	✓
RecordWage	✓	✗	✗	✗	✓	✗
SetPovertyThreshold	✓	✓	✗	✗	✗	✗
QueryWagesByWorker	✓	✓	✓	✗	✓	✓
QueryWagesByEmployer	✓	✓	✓	✗	✓	✗
RegisterUser	✓	✓	✗	✗	✗	✗

```
PS E:\Major-Project>
```

Figure 10. Unit test results showing that our smart contracts and access rules are functioning as intended.

H. Full-Scale Functional Testing

To see how the system would hold up in a real-world scenario, we ran 156 different test cases. These covered 26 different smart contract functions being hit by six different user identities at the same time.

As shown in Fig. 11, we hit a 100% pass rate, which gives us a lot of confidence in the system's stability and reliability.

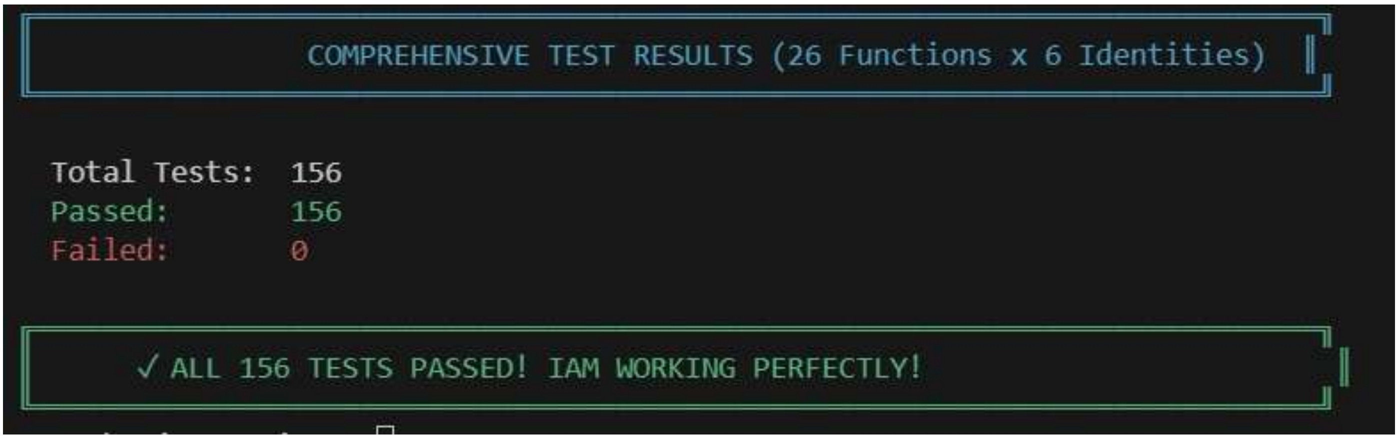


Figure 11. summary of our large-scale testing across all user roles and system functions.

I. Summary of Results

The data from our experiments shows that TRACIENT actually does what it sets out to do. It successfully:

- 1. Uses blockchain to keep wage records permanent and honest,
- 2. Integrates secure QR payments into the workflow,
- 3. Uses AI to profile households fairly and accurately,
- 4. Enforces strict security through smart contracts, and
- 5. Stays stable even under heavy, multi-user testing.

Overall, these results prove that a tech-driven approach to welfare is not just possible, but much more efficient than the traditional methods used today.

VII. CONCLUSION

In this paper, we introduced TRACIENT as a practical solution to the long-standing problem of tracking informal income for welfare distribution. By combining blockchain with AI, we have shown that it is possible to move away from the errors of manual surveys and toward a system built on verified data. Our testing proved that we can record wages immutably, handle secure payments via QR codes, and use machine learning to create fair, accurate household profiles. The successful stress-testing of our smart contracts and the 100% pass rate in functional tests show that the system is not just a concept, but a stable platform ready for real-world use. Ultimately, this framework provides a way to stop the misuse of welfare schemes and ensures that government help reaches the people who actually need it. We believe TRACIENT has a lot of potential as a pilot program for modernizing how we assess income and monitor social safety nets in real-time.

ACKNOWLEDGMENT

We would like to sincerely thank St. Thomas College of Engineering and Technology for their support. The resources provided through the Research and Development Lab were essential to the success of this project and gave us the environment we needed to turn this research into a working reality.

REFERENCES

[1] V. Tyagi and Y. Beri, "Blockchain-integrated digital payment systems in the public sector: innovation, transparency, and economic efficiency in emerging markets," Indian Council of Social Science Research, Vasanta College for Women (BHU), Tech. Rep., 2025.
[2] O. Kuznetsov, E. Frontoni, P. Sernani, L. Romeo, and A. Mancini, "On the integration of artificial intelligence and blockchain technology: a perspective about security," 2025.
[3] T. Bhardwaj and K. Sumangali, "An explainable federated blockchain framework with privacy-preserving AI optimization for securing healthcare data," unpublished, 2025.
[4] B. John, "Blockchain-based KYC and access verification for financial institutions," Jun. 2025.

[5] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, et al., "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," in *Proceedings of the Thirteenth EuroSys Conference (EuroSys '18)*, Porto, Portugal, Apr. 2018.

[6] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*, San Francisco, CA, USA, Aug. 2016

[7] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[8] P. Datta and C. Chaudhuri, "Fragmentation and Informalisation in Indian Labour Market: Analysis Using PLFS Data 2017 to 2020," *Indian Journal of Labour Economics*, 2020.

[9] M. Bordoloi, M. H. Farooqui, and S. Pandey, *Social Security for Informal Workers in India: Exploring India's Labour Market Policies on Provisioning of Social Security to Informal Workers in the Unorganised Sector*. Centre for Policy Research & Accountability Initiative, Nov. 2020.

[10] https://mahafood.gov.in/en/national-food-security-act-2013/

[11] M. Mahamallik and G. B. Sahu, "Identification of the Poor: Errors of Exclusion and Inclusion," *Economic and Political Weekly*, 2013.

[12 International Monetary Fund, *Measuring the Informal Economy*, IMF Policy Paper, Washington, D.C., Feb. 2021.]

[13] Citation: Das P.K. (2024) Taxability of Agricultural Income in India: A Study, British Journal of Multidisciplinary and Advanced Studies: Agriculture, 5(2),13-26

[14] Press Information Bureau, Government of India, "Provisional Data of Socio Economic and Caste Census (SECC) 2011 for Rural India Released," New Delhi, July 3, 2015.