MongoDB/Firebase Project- Team Armaan & Gang

Team Members:

1. Pratima Vanakhade
2. Armaan Mulani
3. Vivek Adlagatta
4. Yash Khapre

Submission Date: 16/09/25

Cohort: Jensen Huang

Live Demo: https://armaanm08.github.io/Fraud_and_Survey/(Only UI since deployed using Github)

Github Link: https://github.com/ArmaanM08/Fraud_and_Survey.git

Topic: Fraud Detection in Digital Payments

Description: Survey users about payment security fears. Store real-time transaction logs in a NoSQL database and analyze them for anomaly detection and fraud patterns.

Project Overview

This report outlines the design and development of our prototype for **payment security and fraud monitoring**. We combine real user surveys with a real-time transaction logging system to detect anomalies in online payments. The web app integrates **Node.js (Express)**, **MongoDB Change Streams**, **Server-Sent Events**,

and **Chart.js** to deliver live dashboards and insights. All work is original; AI was used only for reference (e.g., schema best practices).

Phase 1: Survey Design & Analysis (15%)

Goal:

Understand student and staff concerns about payment security, their current practices, and expectations for fraud alerts.

Platform & Circulation:

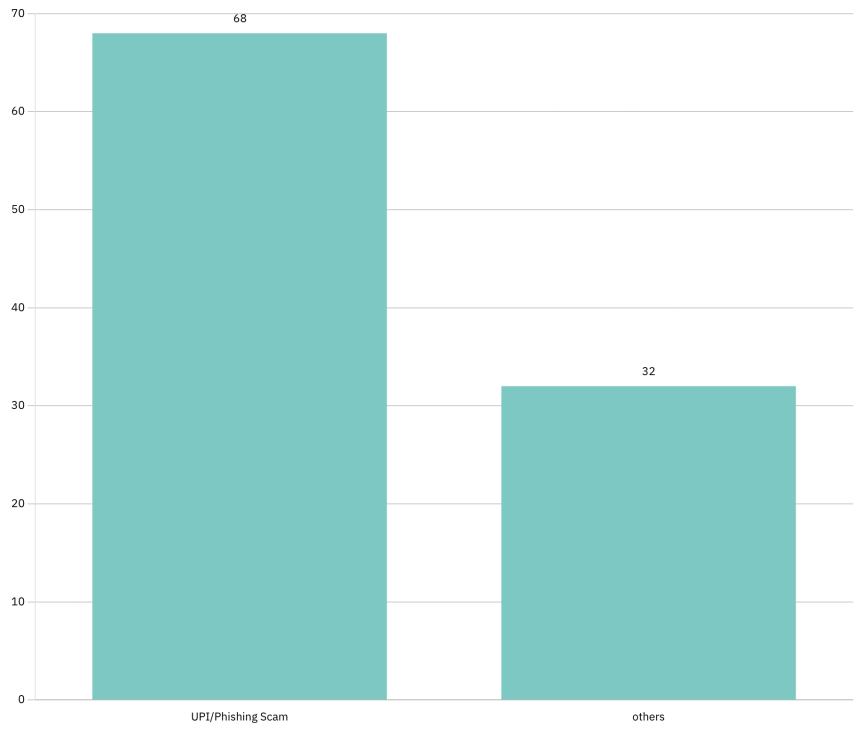
Google Forms survey shared via campus WhatsApp groups and university email. Achieved 30+ valid responses.

Survey Structure:

- **Demographics**: Age group, student/staff status
- **Payment Methods**: UPI, Cards, Wallets, Net Banking
- **Security Fears**: Phishing, UPI fraud, card cloning
- **Security Practices**: OTP, biometrics, app locks
- **Expectations**: Real-time alerts, AI-based detection
- **Past Fraud Experience**: Yes/No, details
- **Open Feedback**: Improvements desired

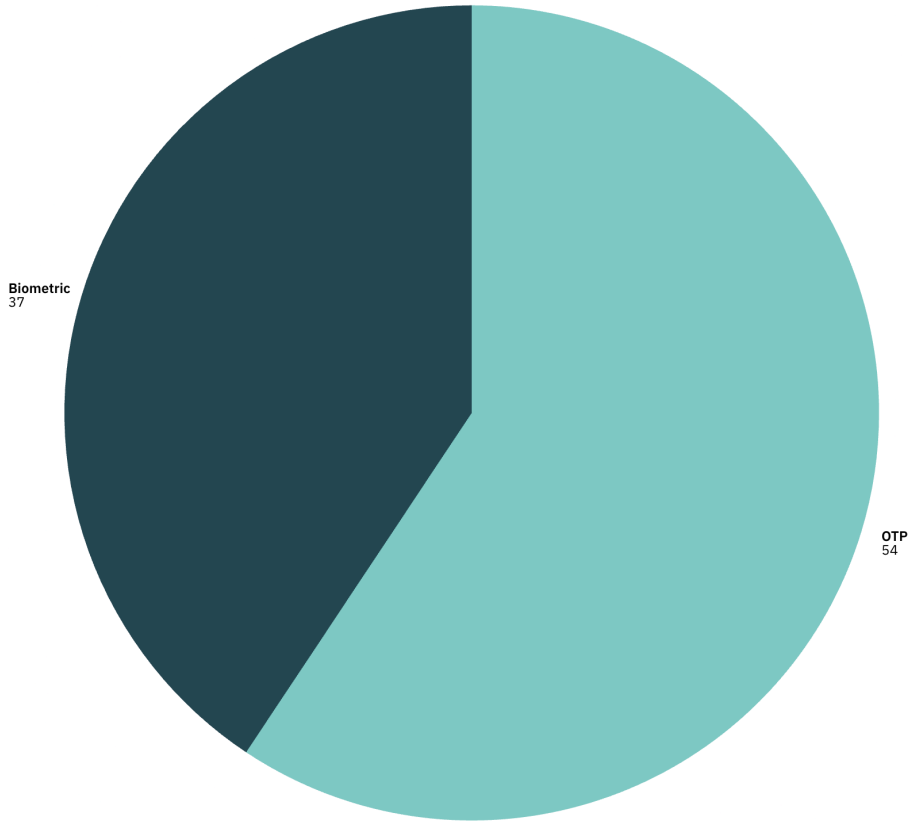
Key Findings (examples):

Security Fears



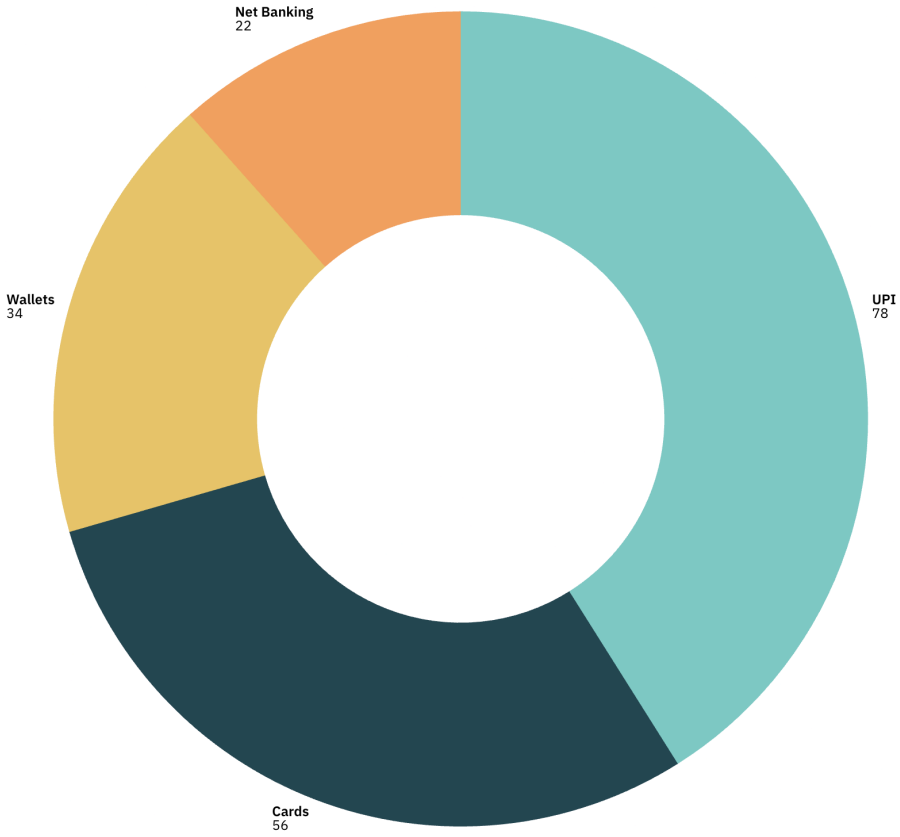
Phishing/UPI fraud as top fear.	68%

Security Practices



Rely on OTP	54%
Rely on Biometrics	37%

Payment Methods



Phase 2: Case Study & Stakeholder Insight (15%)

Selected Process:

Online campus payments (fees, events, cafeteria). Manual review with limited visibility for suspicious activity.

Stakeholder Interviews:

- **Cashier/Accounts Staff:** Reports chargeback disputes and manual review overhead.
- **Students:** Fear unauthorized UPI requests; want clear alerts and blocks on unusual activity.

Current Workflow Issues:

- No automated anomaly checks; manual log review.
- Limited visibility for users on suspicious activity.

Opportunities:

- Real-time anomaly detection on transactions.
- Alerts to users/staff and logged evidence for disputes.

Phase 3: Data Model & Explanation (Extended)

- Database: MongoDB
- Collections: users, devices, transactions, fraud_logs, survey_responses, alerts
- Conventions:
- IDs: transaction_id (string, nanoid), user_id (string), device_id (string)
- Timestamps: ISODate
- Currency: amounts in INR as number (2-decimal), store also as paise if high precision is needed

1) Users

- Purpose: reference data for behavior baselines and notifications.
- Schema:

```
{
"_id": "user_15",
"email": "u15@campus.edu",
"phone": "+91-9XXXXXXXXX",
"preferred_channels": ["push", "email"],
"risk_profile": { "avg_amount_30d": 1520.35, "night_txn_rate": 0.04 },
"last_seen_at": ISODate("2025-09-17T10:22:11Z"),
"created_at": ISODate("2025-07-01T12:00:00Z")
}
```

2) Devices

- Purpose: tie device/IP/location fingerprints to users for mismatch rules.

- Schema:

```
//Apply to anomalyDetec...
{
   "_id": "device_21",
   "user_id": "user_15",
   "device_fingerprint": "ios-17.5-iphone13-pro-ud1",
   "label": "iPhone",
   "trusted": true,
   "first_seen_ip": "192.168.1.24",
   "last_seen_ip": "192.168.1.35",
   "last_seen_location": { "city": "Bengaluru", "country": "IN" },
   "created_at": ISODate("2025-08-01T09:20:00Z"),
   "updated_at": ISODate("2025-09-17T10:22:11Z")
}
//Indexes:
   { user_id: 1, device_fingerprint: 1 } unique
   { updated_at: -1 }
```

3) Transactions

- Purpose: source-of-truth for payment activity; optimized for write and rule evaluation.
- Schema (denormalized for speed):

```
//Apply to anomalyDetec...
{
    "_id": "N/A (use transaction_id as unique alt key)",
    "transaction_id": "4h2Z7eKJr89d",
    "user_id": "user_15",
    "timestamp": ISODate("2025-09-17T10:20:30Z"),
    "amount": 4899.00,
    "amount_paise": 489900,
    "currency": "INR",
```

```
    "payment_method": "UPI", *// UPI | CARD | WALLET | NET_BANKING*
    "merchant": { "id": "m_7781", "name": "Campus Canteen" },
    "location": { "city": "Bengaluru", "country": "IN" },
    "device_id": "device_21",
    "ip_address": "192.168.1.24",
    "status": "SUCCESS", *// SUCCESS | FAILED | PENDING | REVERSED*
    "meta": { "channel": "mobile", "app_version": "2.3.1" },
    "ingest_info": { "source": "simulator", "schema_version": 2 }
}
// Indexes:
    { transaction_id: 1 } unique
    { user_id: 1, timestamp: -1 } (time windows per user)
    { device_id: 1, timestamp: -1 }, { ip_address: 1, timestamp: -1 }
// TTL (optional for raw telemetry): none by default; or add { timestamp: 1 } wit
h expireAfterSeconds for high-volume archives
```

4) Fraud Logs

- Purpose: immutable anomaly flags; every entry explains why it was flagged.
- Schema:

```
// Apply to anomalyDetec...
{
    "_id": ObjectId("..."),
    "transaction_id": "4h2Z7eKJr89d",
    "user_id": "user_15",
    "timestamp": ISODate("2025-09-17T10:20:30Z"),
    "anomaly_detected": true,
    "risk_score": 72, *// 0-99*
    "reason": [
    { "code": "HIGH_AMOUNT_OUTLIER", "message": "Amount 4899 > 3x avg
1500" },
    { "code": "ODD_HOUR", "message": "Transaction at 02:30" }
    ],
    "rules_snapshot": { "v": "1.0.0", "weights": { "HIGH_AMOUNT_OUTLIER": 4
```

```
0, "ODD_HOUR": 15 } },
    "actions": [ "ALERT_USER", "REVIEW_QUEUE" ],
    "ingested_at": ISODate("2025-09-17T10:20:31Z")
}
// Indexes:
    { user_id: 1, timestamp: -1 }
    { risk_score: -1, timestamp: -1 }
    { transaction_id: 1 }
```

5) Survey Responses

- Purpose: drive product requirements and dashboard insights.
- Schema:

```
// Apply to anomalyDetec...
{
    "_id": ObjectId("..."),
    "user_id": null,
    "age_group": "25-34",
    "usage_frequency": "daily",
    "payment_methods": ["UPI", "CARD"],
    "security_fears": ["phishing", "upi_fraud", "identity_theft"],
    "past_fraud_experience": { "had_fraud": true, "details": "UPI request sca
m", "chargeback": false },
    "security_practices": ["otp", "biometrics", "transaction_checks"],
    "security_expectations": ["alerts", "ai_fraud_detection"],
    "created_at": ISODate("2025-09-17T09:10:00Z"),
    "ingest_info": { "source": "web_form", "schema_version": 1 }
}
// Indexes:
    { created_at: -1 }
    { age_group: 1 }, { payment_methods: 1 } (multi-key)
    { security_fears: 1 } (multi-key)
```

6) Alerts

- Purpose: user-facing notifications generated from fraud_logs.
- Schema:

```
// Apply to anomalyDetec...
{
    "_id": ObjectId("..."),
    "user_id": "user_15",
    "transaction_id": "4h2Z7eKJr89d",
    "type": "FRAUD_RISK", *// FRAUD_RISK | INFO | ACTION_REQUIRED*
    "title": "Unusual transaction detected",
    "body": "We detected unusual activity on your account...",
    "status": "SENT", *// PENDING | SENT | READ | DISMISSED*
    "channel": "push",
    "created_at": ISODate("2025-09-17T10:20:32Z"),
    "updated_at": ISODate("2025-09-17T10:22:01Z")
}
    Indexes:
      { user_id: 1, created_at: -1 }
      { status: 1, created_at: -1 }
```

Phase 4: Prototype Functionality (30%)

Core Features:

- **Auth:** Basic demo with mock `user_id` (Firebase-ready swap).
- **CRUD:**
  - `POST /api/survey` – Create survey response
  - `GET /api/survey` – List survey responses
  - Transaction simulator inserts into `transactions` (viewable in Compass).
- **Real-Time Feature:**
  - Change Streams detect inserts.
  - Rules flag anomalies and log into `fraud_logs`.
  - Frontend receives live updates via `/api/anomalies/stream` SSE.

- **Dashboard:**
  - Chart.js visualizes survey insights.
  - Live anomaly feed with risk and reasons.

Technology Stack:

- **Backend:** Node.js (Express) + MongoDB
- **Frontend:** HTML/CSS + Chart.js
- **Real-Time:** MongoDB Change Streams + SSE

Phase 5: Deployment & Validation (15%)

- **Replica Set Enabled:** `-replSet rs0` + `rs.initiate()` locally.
- **Validation in Compass:**
  - Collections: `transactions`, `fraud_logs`, `survey_responses`
  - Filters: `risk_score ≥ 50; last 5 minutes; per user_id`
- **Load Testing:** Adjust `SIMULATOR_INTERVAL_MS` to test throughput; verify rule triggers.
- **Hosting Options:**
  - Static frontend on Firebase Hosting
  - API on VM/Render/Railway or Cloud Run proxy

Phase 6: Final Integration & Handover (15%)

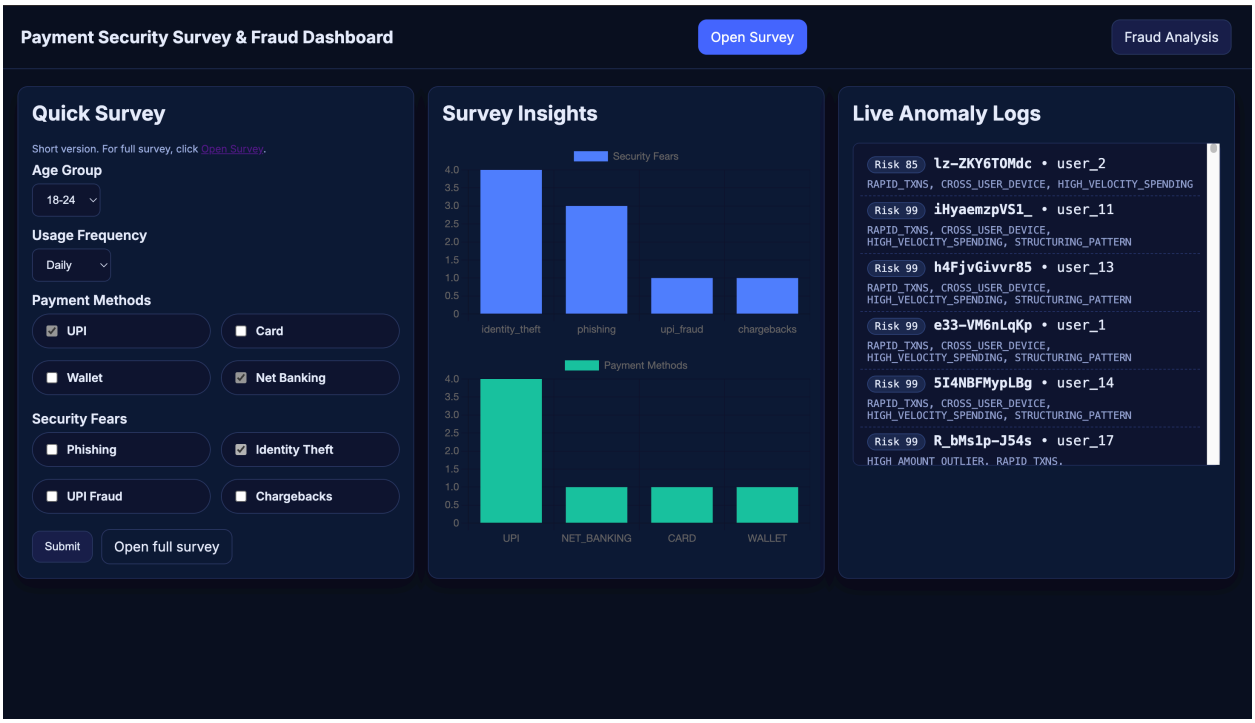
Work Allocation:

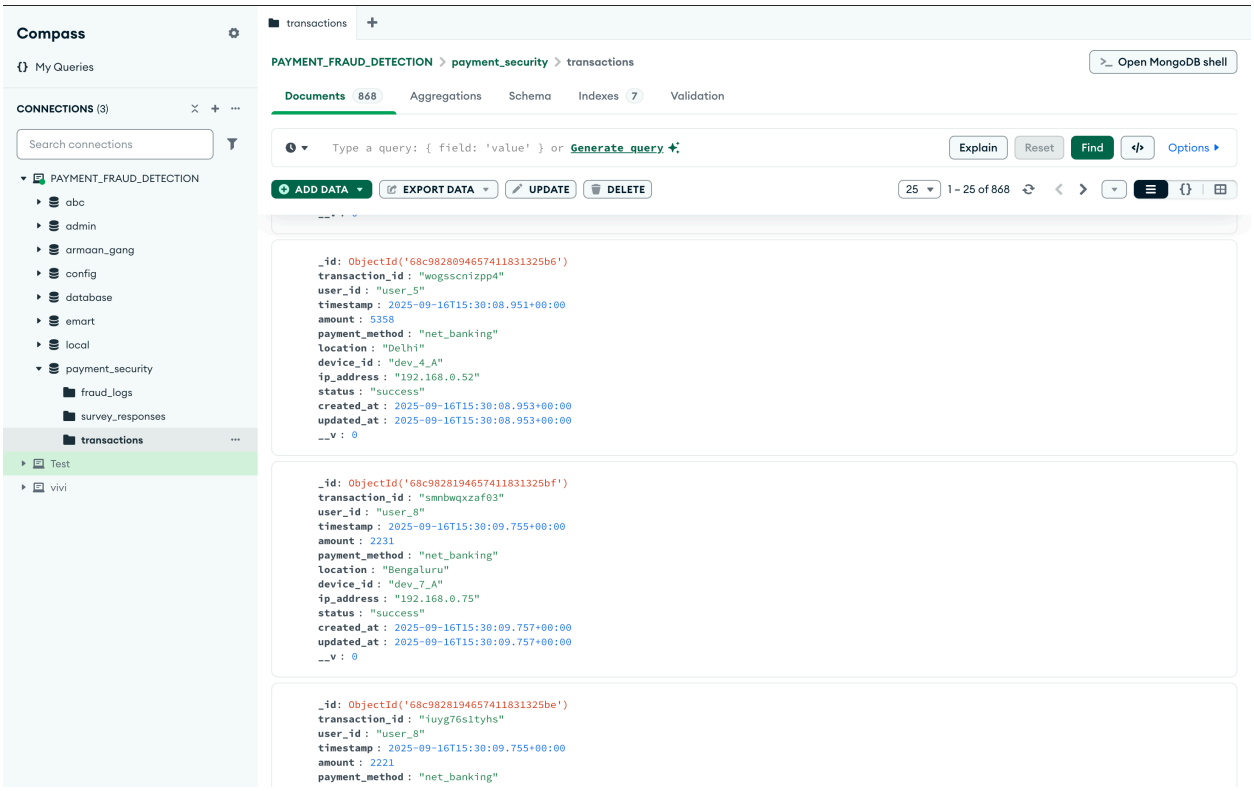
- **Yash:** Performance pass; seed larger datasets; write Operations Guide.
- **Pratima:** Documentation, screenshots of Compass & charts, QA checklist.
- **Armaan:** Frontend polish; accessibility; slide deck.
- **Vivek:** Backend hardening, error handling, demo script.

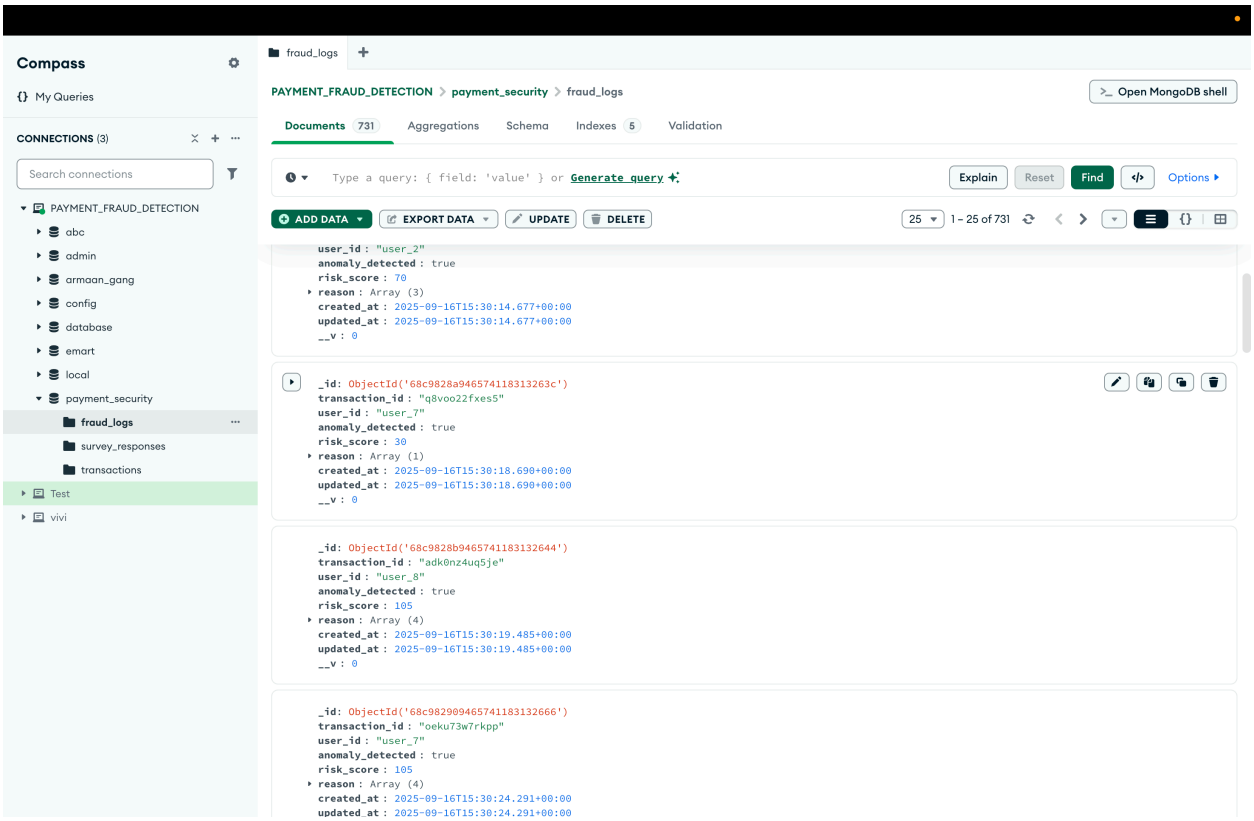
Deliverables:

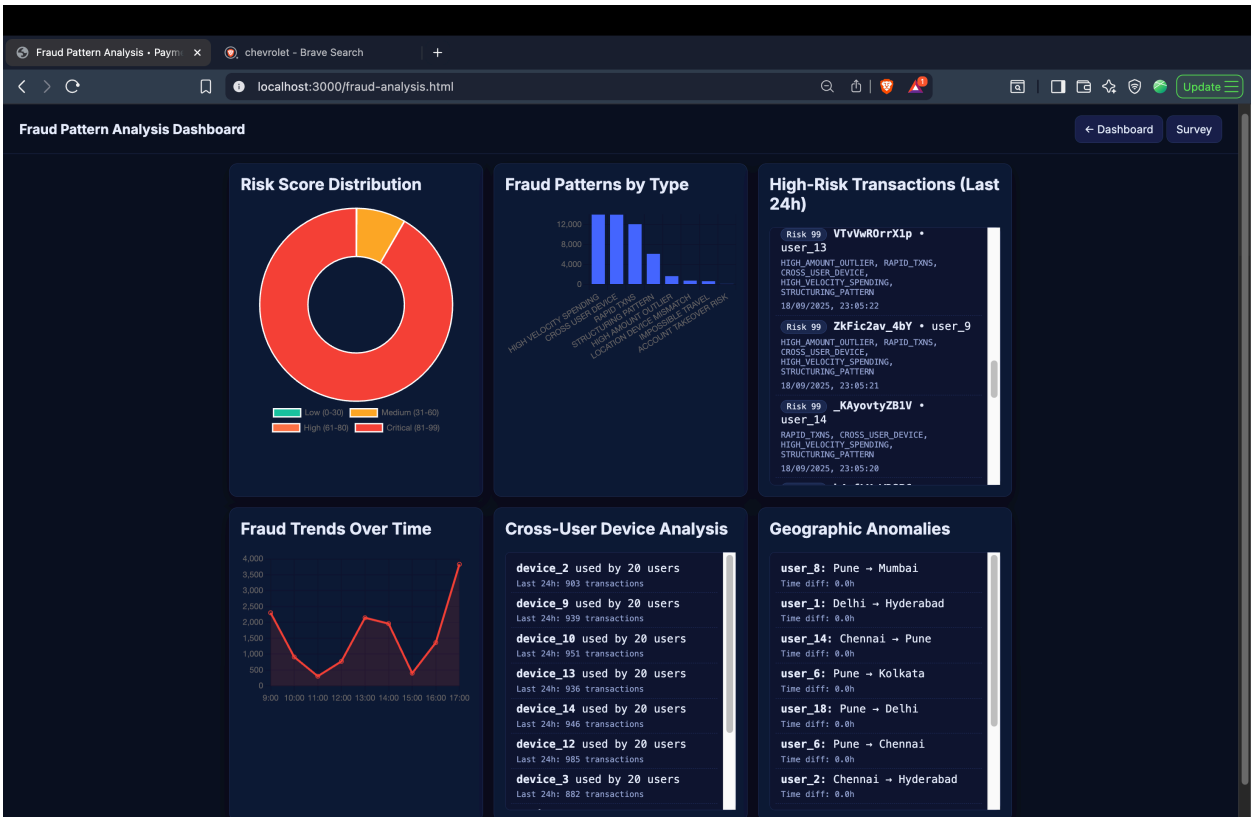
- **Source Code:** Node/Express + MongoDB

- **Report:** Survey analysis, case study, data model, prototype overview
- **Slides + Demo Plan**
- **Operations Guide:** Setup, Compass queries, environment, run commands




Compass
My Queries
CONNECTIONS (3)
Search connections
PAYMENT_FRAUD_DETECTION
abc
admin
armaan_gang
config
database
emart
local
payment_security
fraud_logs
survey_responses
transactions
Test
vivi
transactions
PAYMENT_FRAUD_DETECTION > payment_security > transactions
Open MongoDB shell
Documents 868
Aggregations
Schema
Indexes 7
Validation
Type a query: { field: 'value' } or Generate query
Explain
Reset
Find
Options
ADD DATA
EXPORT DATA
UPDATE
DELETE
25
1 – 25 of 868
_id: ObjectId('68c982809465741183132 5b6')
transaction_id : "wogsscnizpp4"
user_id : "user_5"
timestamp : 2025-09-16T15:30:08.951+00:00
amount : 5358
payment_method : "net_banking"
location : "Delhi"
device_id : "dev_4_A"
ip_address : "192.168.0.52"
status : "success"
created_at : 2025-09-16T15:30:08.953+00:00
updated_at : 2025-09-16T15:30:08.953+00:00
__v : 0
_id: ObjectId('68c9828194657411831325bf')
transaction_id : "smnbwqxzaf03"
user_id : "user_8"
timestamp : 2025-09-16T15:30:09.755+00:00
amount : 2231
payment_method : "net_banking"
location : "Bengaluru"
device_id : "dev_7_A"
ip_address : "192.168.0.75"
status : "success"
created_at : 2025-09-16T15:30:09.757+00:00
updated_at : 2025-09-16T15:30:09.757+00:00
__v : 0
_id: ObjectId('68c9828194657411831325be')
transaction_id : "iuyg76s1tyhs"
user_id : "user_8"
timestamp : 2025-09-16T15:30:09.755+00:00
amount : 2221
payment_method : "net_banking"


Compass
My Queries
CONNECTIONS (3)
Search connections
PAYMENT_FRAUD_DETECTION
abc
admin
armaan_gang
config
database
emart
local
payment_security
fraud_logs
survey_responses
transactions
Test
vivi
fraud_logs
PAYMENT_FRAUD_DETECTION > payment_security > fraud_logs
Open MongoDB shell
Documents 731
Aggregations
Schema
Indexes 5
Validation
Type a query: { field: 'value' } or Generate query
Explain
Reset
Find
Options
ADD DATA
EXPORT DATA
UPDATE
DELETE
25
1 – 25 of 731
user_id : "user_2"
anomaly_detected : true
risk_score : 70
reason : Array (3)
created_at : 2025-09-16T15:30:14.677+00:00
updated_at : 2025-09-16T15:30:14.677+00:00
__v : 0
_id: ObjectId('68c9828a946574118313263c')
transaction_id : "q8voo22fxes5"
user_id : "user_7"
anomaly_detected : true
risk_score : 30
reason : Array (1)
created_at : 2025-09-16T15:30:18.690+00:00
updated_at : 2025-09-16T15:30:18.690+00:00
__v : 0
_id: ObjectId('68c9828b9465741183132644')
transaction_id : "adk0nz4uq5je"
user_id : "user_8"
anomaly_detected : true
risk_score : 105
reason : Array (4)
created_at : 2025-09-16T15:30:19.485+00:00
updated_at : 2025-09-16T15:30:19.485+00:00
__v : 0
_id: ObjectId('68c9829094657411831326 66')
transaction_id : "oeku73w7rkpp"
user_id : "user_7"
anomaly_detected : true
risk_score : 105
reason : Array (4)
created_at : 2025-09-16T15:30:24.291+00:00
updated_at : 2025-09-16T15:30:24.291+00:00



Github Repo Link: https://github.com/vivek-419/MONGODB-FIREBASE_ASSIGNMENT