NSHURO ARNAUD NELLIGAN 27960

PL/SQL

PHASE III — LOGICAL DATA MODEL

Security Violation Notifier System

---

1. IDENTIFIED ENTITIES + ATTRIBUTES + DATA TYPES

1.1 USER Entity

Stores all system users whose actions are monitored.

Attribute	Data Type	Description
user_id (PK)	NUMBER(10)	Unique user identifier
full_name	VARCHAR2(100)	Person's name
email	VARCHAR2(100)	Contact
department	VARCHAR2(100)	Work department
role	VARCHAR2(50)	user/admin/security
account_status	VARCHAR2(20)	Active, Locked
last_login	DATE	Last login timestamp

1.2 SYSTEM_LOG Entity

Captures all user actions (normal or suspicious).

Attribute	Data Type
log_id (PK)	NUMBER(10)
user_id (FK)	NUMBER(10)
action_type	VARCHAR2(100)

Attribute	**Data Type**
resource_accessed	VARCHAR2(100)
execution_time	DATE
outcome	VARCHAR2(20)

1.3 VIOLATION Entity

Created when system identifies suspicious activity.

Attribute	**Data Type**
violation_id (PK)	NUMBER(10)
user_id (FK)	NUMBER(10)
violation_type	VARCHAR2(100)
timestamp	DATE
affected_resource	VARCHAR2(100)
severity_level	VARCHAR2(20)
ip_address	VARCHAR2(50)
description	VARCHAR2(500)
status	VARCHAR2(20)

1.4 ALERT Entity

Notification sent to admin when violation occurs.

Attribute	**Data Type**
alert_id (PK)	NUMBER(10)
violation_id (FK)	NUMBER(10)

Attribute	Data Type
sent_to (FK → admin.admin_id)	NUMBER(10)
delivery_channel	VARCHAR2(30)
delivery_time	DATE
read_status	VARCHAR2(20)

1.5 ADMIN Entity

Admins/security officers who receive alerts.

Attribute	Data Type
admin_id (PK)	NUMBER(10)
full_name	VARCHAR2(100)
email	VARCHAR2(100)
role	VARCHAR2(50)
privilege_level	VARCHAR2(20)

1.6 CASE Entity (Investigation Table)

Attribute	Data Type
case_id (PK)	NUMBER(10)
violation_id (FK)	NUMBER(10)
admin_id (FK)	NUMBER(10)
investigation_notes	VARCHAR2(500)
investigation_time	DATE
final_status	VARCHAR2(20)

2. FULL ERD (TEXT REPRESENTATION WITH CARDINALITIES)

USER (1) -------- (M) SYSTEM_LOG

USER (1) -------- (M) VIOLATION

VIOLATION (1) -------- (1) ALERT

VIOLATION (1) -------- (M) CASE

ADMIN (1) -------- (M) ALERT

ADMIN (1) -------- (M) CASE

Explanation

- A user can produce many logs.
- A user can commit many violations.
- Every violation generates **exactly one alert**.
- A violation may have **multiple investigation cases**.
- An admin may handle many alerts or cases.

3. CONSTRAINTS (REQUIRED FOR PHASE 3)

PK Constraints

- user_id, log_id, violation_id, alert_id, admin_id, case_id

FK Constraints

- system_log.user_id → user.user_id
- violation.user_id → user.user_id
- alert.violation_id → violation.violation_id

- alert.sent_to → admin.admin_id
- case.violation_id → violation.violation_id
- case.admin_id → admin.admin_id

CHECK Constraints

- severity_level IN (‘LOW’, ‘MEDIUM’, ‘HIGH’)
- account_status IN (‘ACTIVE’, ‘LOCKED’)
- outcome IN (‘ALLOWED’, ‘BLOCKED’, ‘FAILED’)
- read_status IN (‘READ’, ‘UNREAD’)

UNIQUE Constraints

- user.email (each user email must be unique)
- admin.email (unique admin email)

4. NORMALIZATION

1NF

- All tables have atomic values.
- No repeating groups.
- No multivalued attributes.

2NF

- No partial dependence (all PKs are single-column keys).
- All non-key attributes fully depend on PK.

3NF

- No transitive dependency.
- For example:
  - Admin email → Admin name is NOT duplicated in ALERT table
  - User department is NOT stored in violation table

Justification:
Each table represents **one concept**, and all attributes depend ONLY on the PK.
There are NO derived attributes, NO repeating groups, NO duplicated data → **3NF achieved**.

---

5. DATA DICTIONARY (FULLY DETAILED AS REQUIRED)

✓ Includes

- table names
- columns
- data types
- purpose
- constraints

If you want, I can generate it as a **PDF, DOCX, or PPTX**.

---

6. BI CONSIDERATIONS (REQUIRED)

Fact Table

➞ FACT_VIOLATION
Contains measurable events (severity, count, frequency).

Dimension Tables

- DIM_USER
- DIM_ADMIN
- DIM_TIME
- DIM_RESOURCE
- DIM_VIOLATION_TYPE

Slowly Changing Dimensions

DIM_USER (role, department may change → SCD Type 2)

Aggregation Levels Needed

- Violations per user per day

- Violations per department
- High severity trends
- Alert response time
- Admin performance (cases resolved)

Audit Trail

SYSTEM_LOG acts as the primary audit table.

7. ASSUMPTIONS

1. Every violation must generate exactly one alert.
2. A violation may have multiple investigation cases.
3. Every user has only one role at a time.
4. Admins have unique emails.
5. Logs are generated 24/7, even when no violations occur.
6. IP address stored as text, not number.
7. System must maintain at least 5 years of logs.