ER Diagram & Normalization Report

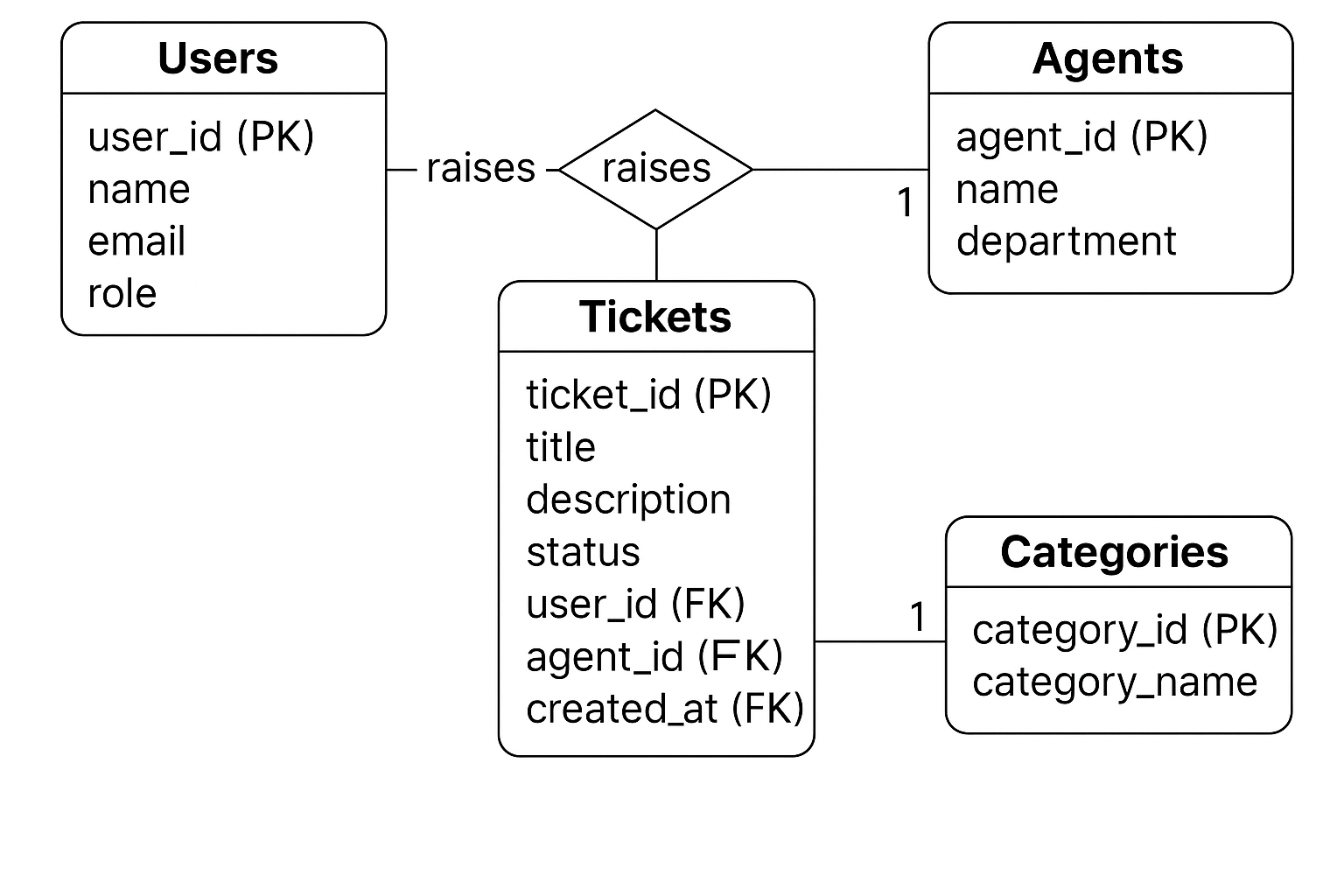
# Entities and Attributes:

|  |  |
| --- | --- |
| *Entity* | *Attributes* |
| User | user\_id (PK), name, email, role |
| Agent | agent\_id (PK), name, department |
| Category | category\_id (PK), category\_name |
| Ticket | ticket\_id (PK), title, description, status, created\_at, user\_id (FK), agent\_id (FK), category\_id (FK) |

# Relationships:

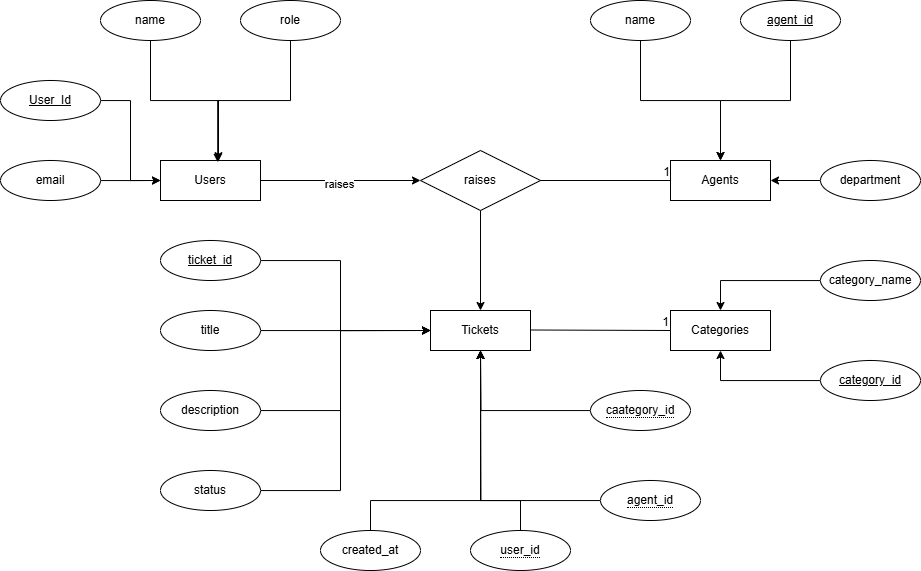
- A User can raise many Tickets  
- An Agent can handle many Tickets  
- A Category can classify many Tickets  
- A Ticket belongs to one User, one Agent, and one Category

**Entity-Relationship Diagram:**



* A User raises a Ticket.
* Each Ticket is assigned to one Agent and categorized under one Category.
* The created\_at field records the timestamp when the ticket is raised.
* Relationships:
  + **One-to-Many** from **User → Tickets**
  + **One-to-Many** from **Agent → Tickets**
  + **One-to-Many** from **Category → Tickets**

**Expanded Entity-Relationship (ER) Diagram – Attribute-Level View:**



* The diagram provides an attribute-level breakdown for all entities.
* Users (identified by user\_id) raise multiple Tickets, each containing title, description, and status.
* Each Ticket is assigned to a specific Agent (by agent\_id) and is categorized under a Category (category\_id).
* The created\_at attribute tracks the exact timestamp a ticket is created.
* Cardinality:
  + **Users ↔ Tickets**: One user can raise many tickets.
  + **Agents ↔ Tickets**: One agent can be responsible for many tickets.
  + **Categories ↔ Tickets**: One category can apply to many tickets.

Normalization Report – Smart Service Desk

# 1. Unnormalized Form (UNF):

This is the initial unnormalized structure where multiple entities' data is stored in one table, resulting in redundancy and anomalies.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *TicketID* | *Title* | *Description* | *Status* | *CreatedAt* | *UserID* | *UserName* | *Email* | *AgentID* | *AgentName* | *Department* | *CategoryID* | *CategoryName* |
| 1 | Login Issue | Can't login | Open | 2025-06-10 10:00:00 | 101 | Karthik | karthik@domain.com | 201 | John | IT | 301 | Software Issue |
| 2 | Keyboard Fault | Keyboard not working | Open | 2025-06-10 11:00:00 | 102 | Sai | sai@domain.com | 202 | Alice | Hardware | 302 | Hardware Issue |
| 3 | Reset Issue | Forgot password | Closed | 2025-06-10 12:00:00 | 101 | Karthik | karthik@domain.com | 201 | John | IT | 301 | Software Issue |

# 2. First Normal Form (1NF):

In 1NF, we ensure each field contains atomic values and remove repeating groups. The table is already in 1NF as each attribute holds atomic values.

# 3. Second Normal Form (2NF)

In 2NF, partial dependencies are removed by dividing the data into logical tables.

## Users Table

|  |  |  |
| --- | --- | --- |
| *UserID* | *Name* | *Email* |
| 101 | Karthik | karthik@domain.com |
| 102 | Sai | sai@domain.com |

## Agents Table

|  |  |  |
| --- | --- | --- |
| *AgentID* | *Name* | *Department* |
| 201 | John | IT |
| 202 | Alice | Hardware |

## Categories Table

|  |  |
| --- | --- |
| *CategoryID* | *CategoryName* |
| 301 | Software Issue |
| 302 | Hardware Issue |

## Tickets Table

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *TicketID* | *Title* | *Description* | *Status* | *CreatedAt* | *UserID* | *AgentID* | *CategoryID* |
| 1 | Login Issue | Can't login | Open | 2025-06-10 10:00:00 | 101 | 201 | 301 |
| 2 | Keyboard Fault | Keyboard not working | Open | 2025-06-10 11:00:00 | 102 | 202 | 302 |
| 3 | Reset Issue | Forgot password | Closed | 2025-06-10 12:00:00 | 101 | 201 | 301 |

# 4. Third Normal Form (3NF)

No transitive dependencies exist. Each non-key attribute is only dependent on the primary key. The schema is now in 3NF.

# 5. Summary of Normalization Steps

|  |  |  |
| --- | --- | --- |
| *Normal Form* | *Action Taken* | *Outcome* |
| UNF | Combined entity data with redundancy | Inefficient and prone to anomalies |
| 1NF | Atomic values ensured, no repeating groups | Satisfies atomic structure |
| 2NF | Removed partial dependencies | Efficient, reduced redundancy |
| 3NF | Removed transitive dependencies | Fully normalized relational schema |