Group 8 Team Members

- Nikhila B R
- Han-Sheng Chen
- Laksh Dhamija
- Harshally Maruti Mutgekar

Banking Management System Logical ERD

This documentation explains the steps taken to convert the conceptual Entity-Relationship Diagram (ERD) into a logical ERD, with additional clarifications and enhancements for better understanding.

Steps Taken in Conversion

1. **Entity to Table Conversion**
    - Each entity in the conceptual ERD was converted into a table.
    - Attributes of entities were used as columns in their respective tables.

2. **Primary and Foreign Keys**
    - Primary keys (PK) were defined for all tables to uniquely identify each record.
    - Foreign keys (FK) were added to establish relationships between tables. For example:
        - `branch_id` is a foreign key in the `Customer` table.
        - `customer_id` is a foreign key in the `Account` table.

3. **Composite Attributes Simplified**
    - Composite attributes were broken down into simple attributes.
        - Example: The `address` attribute in the `Customer` entity was split into `address_line_1`, `address_line_2` and `pincode`.

4. **Weak Entities Conversion**
    - Weak entities were converted into independent tables by combining their partial identifier with the primary key of their associated strong entity as a composite primary key.
        - Example: The `Transaction` weak entity was turned into a table with `transaction_id` (partial identifier) and `strong_entity_id` (foreign key) as its composite primary key.

5. **Relationships Representation**
    - **One-to-Many Relationships:** The primary key of the "one" side was added as a foreign key on the "many" side.

        - Example: The relationship between `Branch` and `Customer` is represented by adding `branch_id` as a foreign key in the `Customer` table.
    - **Many-to-Many Relationships:** A separate associative table was created, with its primary key being a composite of foreign keys from both participating entities.
        - Example: The many-to-many relationship between `Customer` and `Financial_Instrument` resulted in the creation of the `Customer_Financial_Insurance` table.
    - **Unary Relationships:** For recursive relationships, foreign keys were added within the same table.
        - Example: In the `Employee` table, `manager_id` serves as a recursive foreign key to represent managerial hierarchy.

6. **Enhanced EER Representation**
    - Subtypes and supertypes were handled using separate tables, with subtype discriminators added to the supertype table.
        - Example: The `Insurance` table serves as a supertype for subtypes like `Health Insurance`, `Property Insurance`, and `Vehicle Insurance`. A discriminator column (`insurance_type`) identifies the specific subtype.
        - The `Loan` table serves as a supertype for subtypes like `Vehincle_loan`, `Student_loan`, `mortgage_loan` and `business_loan`.
        - The `Financial Instrument` table serves as a supertype for subtypes like `Loan`, `Insurance`, `Credit Card` and `Account`.

7. **Normalization to 3NF**
    - All tables were normalized to Third Normal Form (3NF) by ensuring:
        - No composite or multi-valued attributes exist.
        - No partial or transitive dependencies.

Key Design Features

- **Associative Entities:**
  Associative entities like `Customer_Financial_Instrument` were introduced to handle complex relationships while maintaining normalization standards.
- **Subtype Discriminators:**
  Discriminators like `loan_type`, `insurance_type` and `instrument_type` were added to supertype tables (`Loan`, `Insurance`, `Financial_instrument`) to differentiate between subtypes.

- **Recursive Relationships:**
  Recursive relationships, such as employees managing other employees, are captured using self-referencing foreign keys (`manager_id`) in the same table.

Additional Enhancements

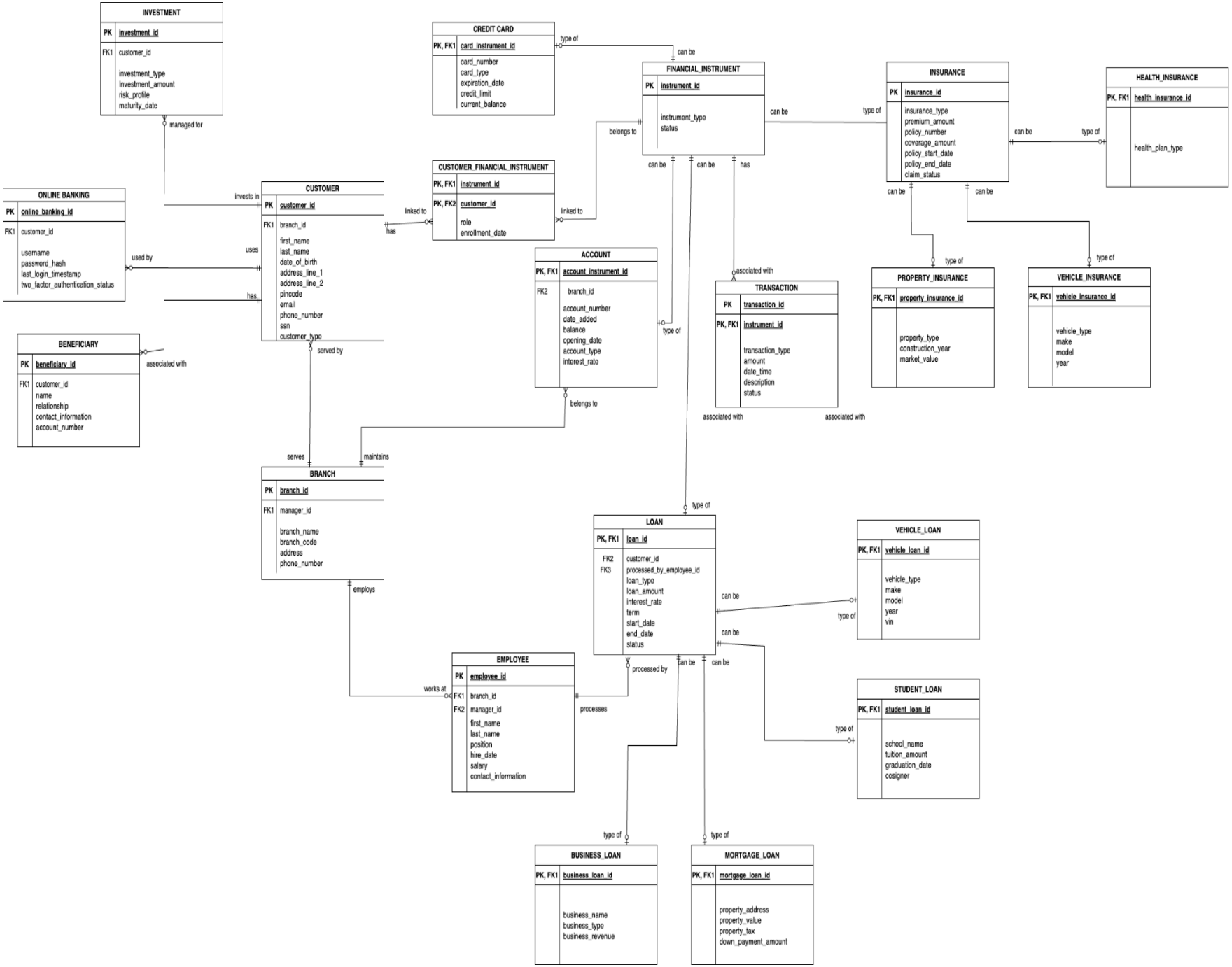
1. **Transaction Table Details:**
   The `Transaction` table includes:
   - Attributes like `transaction_type`, `amount`, and timestamps (`date_time`) for better tracking.
   - Associations with both accounts and strong entities for comprehensive linkage.

2. **Loan Subtypes:**
   Loans are categorized into subtypes (`Vehicle Loan`, `Student Loan`, etc.), each with unique attributes:
   - Vehicle loans include details like vehicle type, make, model, year, and VIN.
   - Student loans include school name, tuition amount, graduation date, and cosigner details.

3. **Online Banking Integration:**
   The inclusion of an `Online Banking` table captures digital banking features such as:
   - Login credentials (`username`, hashed password).
   - Two-factor authentication status for enhanced security.

4. **Branch Management:**
   Each branch maintains its own employees (`Employee`) and customers (`Customer`). This hierarchical structure is reflected through relationships between the tables.

5. **Financial Instruments Details:**
   Financial Instruments are linked to customers via an associative entity (`Customer_Financial_Instruments`) and further detailed in subtypes like loan, account, credit cards and insurance.


INVESTMENT
PK investment_id
FK1 customer_id
investment_type
Investment_amount
risk_profile
maturity_date
managed for
CREDIT CARD
PK, FK1 card_instrument_id
card_number
card_type
expiration_date
credit_limit
current_balance
type of
can be
FINANCIAL_INSTRUMENT
PK instrument_id
instrument_type
status
belongs to
can be
type of
INSURANCE
PK insurance_id
insurance_type
premium_amount
policy_number
coverage_amount
policy_start_date
policy_end_date
claim_status
can be
type of
HEALTH_INSURANCE
PK, FK1 health_insurance_id
health_plan_type
ONLINE BANKING
PK online_banking_id
FK1 customer_id
username
password_hash
last_login_timestamp
two_factor_authentication_status
used by
has
invests in
uses
CUSTOMER
PK customer_id
FK1 branch_id
first_name
last_name
date_of_birth
address_line_1
address_line_2
pincode
email
phone_number
ssn
customer_type
linked to
has
CUSTOMER_FINANCIAL_INSTRUMENT
PK, FK1 instrument_id
PK, FK2 customer_id
role
enrollment_date
linked to
ACCOUNT
PK, FK1 account_instrument_id
FK2 branch_id
account_number
date_added
balance
opening_date
account_type
interest_rate
type of
belongs to
TRANSACTION
PK transaction_id
PK, FK1 instrument_id
transaction_type
amount
date_time
description
status
associated with
associated with
associated with
PROPERTY_INSURANCE
PK, FK1 property_insurance_id
property_type
construction_year
market_value
type of
VEHICLE_INSURANCE
PK, FK1 vehicle_insurance_id
vehicle_type
make
model
year
type of
BENEFICIARY
PK beneficiary_id
FK1 customer_id
name
relationship
contact_information
account_number
associated with
served by
serves
maintains
BRANCH
PK branch_id
FK1 manager_id
branch_name
branch_code
address
phone_number
employs
LOAN
PK, FK1 loan_id
FK2 customer_id
FK3 processed_by_employee_id
loan_type
loan_amount
interest_rate
term
start_date
end_date
status
type of
can be
can be
can be
can be
VEHICLE_LOAN
PK, FK1 vehicle_loan_id
vehicle_type
make
model
year
vin
type of
STUDENT_LOAN
PK, FK1 student_loan_id
school_name
tuition_amount
graduation_date
cosigner
type of
EMPLOYEE
PK employee_id
FK1 branch_id
FK2 manager_id
first_name
last_name
position
hire_date
salary
contact_information
works at
processes
processed by
BUSINESS_LOAN
PK, FK1 business_loan_id
business_name
business_type
business_revenue
type of
MORTGAGE_LOAN
PK, FK1 mortgage_loan_id
property_address
property_value
property_tax
down_payment_amount
type of