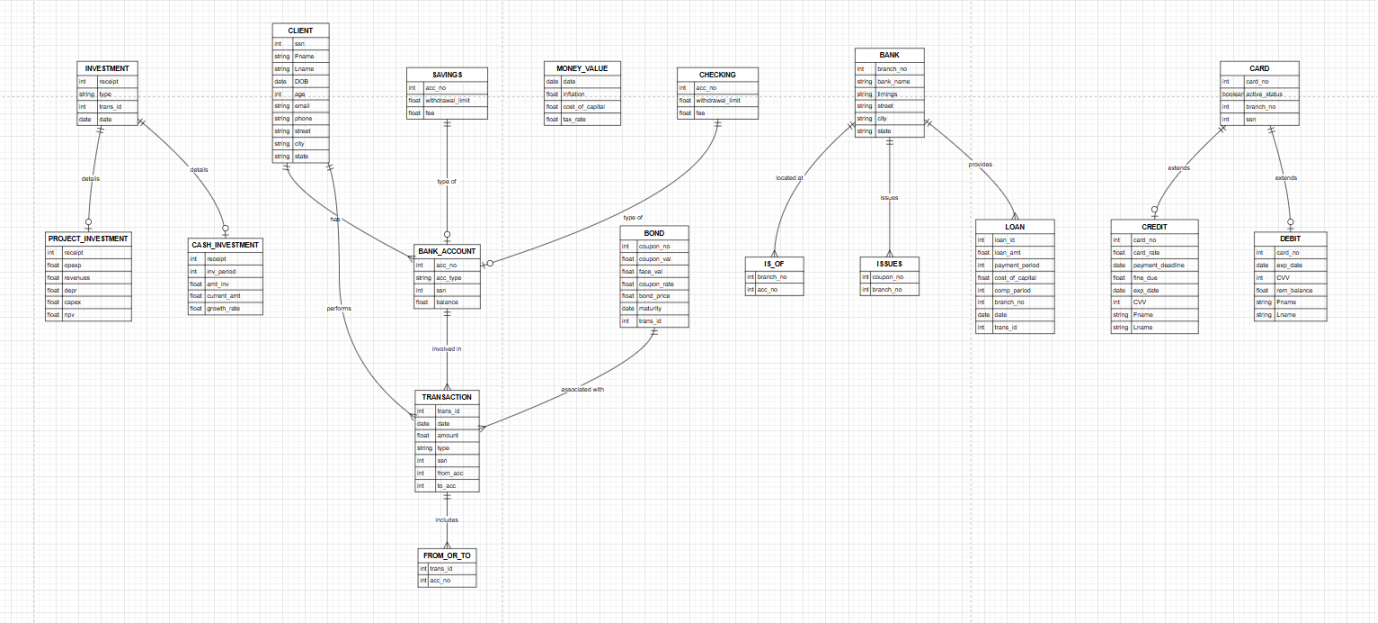
Database systems

Name- Arya Goyal & Gitesh Chinawalkar
Date- 15/11/2024

DBMS PROJECT Part 2

Total in points-
Professor's Comments-



The ER model has been converted into a relational schema following standard conventions:

- The CLIENT-to-TRANSACTION (1:N) relationship is represented by adding the **SSN** as a foreign key in the TRANSACTION table.
- The CLIENT-to-BANK_ACCOUNT (1:N) relationship is handled similarly, with the **SSN** added as a foreign key in the BANK_ACCOUNT table.
- For the TRANSACTION-to-BANK_ACCOUNT (M:N) relationship, a new table called FROM_OR_TO is created, using **trans_id** and **acc_no** as primary keys.
- The TRANSACTION-to-INVESTMENT (1:1) relationship is mapped by adding **trans_id** as a foreign key in the INVESTMENT table.
- The TRANSACTION-to-LOAN (1:1) relationship is implemented with **trans_id** as a foreign key in the LOAN table.
- The TRANSACTION-to-BOND (1:1) relationship uses **trans_id** as a foreign key in the BOND table.
- The USES relationship between MONEY_VALUE and INVESTMENT, LOAN, and BOND tables is established by adding the **date** attribute as a foreign key in these tables.
- The BANK_ACCOUNT-to-BANK (M:N) relationship is mapped through a new table called IS_OF, with **acc_no** and **branch_no** as its primary keys.
- The BANK-to-CARD (1:N) relationship is represented by including **branch_no** as a foreign key in the CARD table.
- The CARD-to-CLIENT (1:N) relationship is defined by adding **SSN** as a foreign key in the CARD table.
- BANK_ACCOUNT types (SAVINGS and CHECKING) are linked to the BANK_ACCOUNT table using **acc_no** as a foreign key in each type-specific table.

- INVESTMENT types (PROJECT_INVESTMENT and CASH_INVESTMENT) are connected to the INVESTMENT table through the **receipt** foreign key.
- The BANK-to-BOND (M:N) relationship is managed through an ISSUES table, which uses **acc_no** and **coupon_no** as primary keys.
- The BANK-to-LOAN (1:N) relationship is handled by adding **branch_no** as a foreign key in the LOAN table.
- The CARD types (CREDIT and DEBIT) are linked back to the CARD table using **card_no** as a foreign key in both tables.

This structure ensures the relationships between entities are preserved in the relational model while maintaining referential integrity.