**COMP3005 Project Report**

Tom Lam – 101114541

**Assumptions:**

Cardinality:

* Participation type is indicated by the number of participants next to each entity in the ER diagram. Specifically, “1..\*” or “1” means total participation while “0..\*” or “0..1” means partial participation
* A book can be published by exactly 1 publisher, A publisher can publish 1 to many books
* A book can be collected by 0 to 1 owner, an owner can collect 0 to many book. There’s no book that belongs to 2 different owner
* A book can be ordered by 0 to many orders, an order can order 1 to many book
* An order can be placed by exactly 1 customer, a customer can place 0 to many order

Other:

* Different authors have different name (to avoid the use of author\_id)
* Different publishes have different name (to avoid the use of publisher\_id)
* Different customer has different name (to avoid the use of customer\_id)

**ER Diagram**

Diagram

Description automatically generated

**Relational Schema**

Book(ISBN, publisher\_name, book\_name, genre, description, num\_of\_pages, price)

Author(ISBN, name)

Publisher(name, email, bank\_account, balance, address, phone\_number)

Owner(name, bank\_account, balance, address, email, phone\_number)

Collect(ISBN, owner\_name, unit\_in\_stock, unit\_sold, revenue, expense, profit, publisher\_split)

Order(order\_id, billing\_address, shipping\_address, status, ordered\_date, estimated\_arrivate, location)

Customer(name, bank\_account, balance, billing\_address, shipping\_address)

CustomerOrder(order\_id, customer\_name)

OrderBook(ISBN, order\_id, unit\_ordered)

**Normalization**

Functional dependencies

Book(ISBN, publisher\_name, book\_name, genre, description, num\_of\_pages, price)

ISBN 🡪 name, publisher\_name, book\_name, genre, description, num\_of\_pages, price

Author(ISBN, name)

ISBN, name 🡪 ISBN, name

Publisher(name, email, bank\_account, balance, address, phone\_number)

name 🡪 email, bank\_account, balance, address, email, phone\_number

Owner(name, bank\_account, balance, address, email, phone\_number)

name 🡪 bank\_account, balance, address, email, phone\_number

Collect(ISBN, owner\_name, unit\_in\_stock, unit\_sold, revenue, expense, profit, publisher\_split)

ISBN 🡪 owner\_name, unit\_in\_stock, unit\_sold, revenue, expense, profit, publisher\_split

Order(order\_id, billing\_address, shipping\_address, status, ordered\_date, estimated\_arrivate, location)

order\_id 🡪 billing\_address, shipping\_address, status, ordered\_date, estimated\_arrivate, location

Customer(name, bank\_account, balance, billing\_address, shipping\_address)

name 🡪 bank\_account, balance, billing\_address, shipping\_address

CustomerOrder(order\_id, customer\_name)

order\_id 🡪 customer\_nameßœ

OrderBook(ISBN, order\_id, unit\_ordered)

ISBN, order\_id 🡪 unit\_ordered

All relations are already in BCNF because their functional dependency is also the superkey

**Schema Diagram**

Diagram

Description automatically generated