COMP 3005: Database Management Systems **(Due: Dec. 10th, 2021 (11:59 PM))**

COMP3005 Final Project Report

Instructor: Ahmed El-Roby *Name: , ID:*

Introduction

This is the project report for the COMP3005A Final Project for the Fall 2021 term.
The group for this project consists of the following members...

Group Members

- Aaron Buitenwerf ()
- Hadi Cheaito ()
- Nabeel Warsalee (101103167)

All project files and source code can be found at the following Github repository...

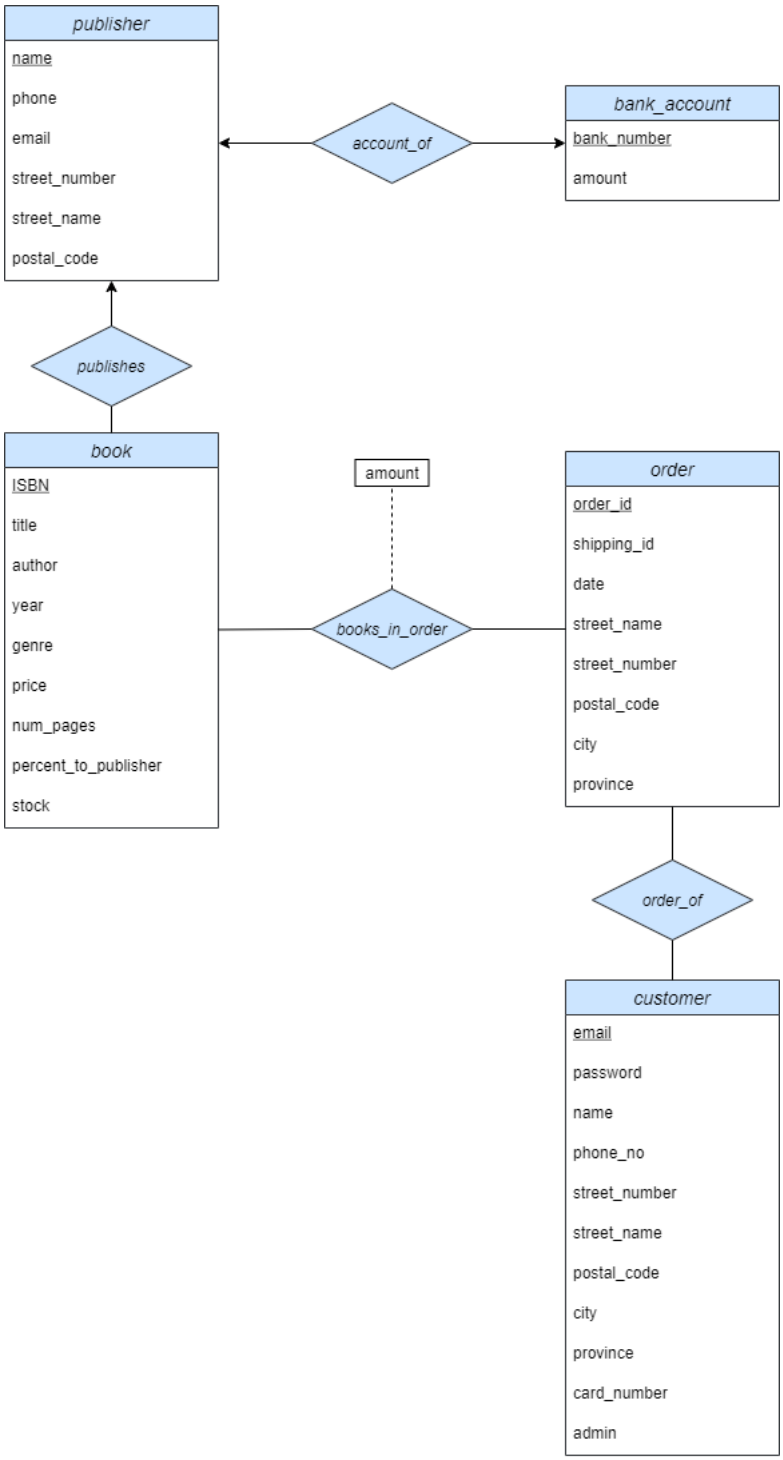
1 Conceptual Design

Insert preamble about design.

Assumptions Made
In this section we list all the assumptions that were made for certain aspects of the problem statement. These assumptions reflect how we designed and organized our database for this project.

- A book can only have one publisher
- All books with same title have the same ISBN
- Assume an order can only have one of the same book (i.e., user cannot buy two copies of the same book)
- Each publisher has only one bank account
- There is only one report made per publisher

ER Diagram
The following is the Entity-Relationship Diagram created to model the entities and relationships from the provided problem statement using the assumptions we have outlined above.



2 Reduction to Relation Schemas

Here are the relation schemas gained from reducing our ER diagram into relations... (Note: Primary keys are underlined)

book(isbn, title, author, genre, year, price, num_pages, publisher_name, stock, percent_to_publisher)
order(order_id, email, shipping_id, date, street_number, street_name, postal_code, city, province)
books_in_order(order_id, isbn, amount)
customer(email, password, name, phone, street_number, street_name, postal_code, city, province, card_number, admin)

publisher(name, phone, bank_number, email, street_number, street_name, postal_code)
bank_account(bank_number, amount)

3 Normalization of Relation Schemas

Functional Dependencies

book
isbn → title, author, genre, year, price, num_pages, publisher_name, stock, percent_to_publisher
title, author → isbn, genre, year, price, num_pages, publisher_name, stock, percent_to_publisher

order
order_id → email, shipping_id, date, street_number, street_name, postal_code, city, province
shipping_id → email, order_id, date, street_number, street_name, postal_code, city, province
postal_code → city, province

books_in_order
order_id, isbn → amount

customer
email, password → name, phone, street_number, street_name, postal_code, city, province, card_number, admin
postal_code → city, province

publisher
name → email, phone, bank_number, street_number, street_name, postal_code
email → name, phone, bank_number, street_number, street_name, postal_code

bank_account
bank_number → amount

Good Normal Form Check and Decomposition

book

1st FD...
Closure of *isbn*, (*isbn*)+ = (isbn, title, author, genre, year, price, num_pages, publisher_name, stock, percent_to_publisher)
Since the closure of isbn includes all attributes in the relation, it means isbn is a superkey for the relation and it complies with BCNF.

2nd FD...
Closure of *title, author*, (*title, author*)+ = (isbn, title, author, genre, year, price, num_pages, publisher_name, stock, percent_to_publisher)
Since the closure of (title, author) includes all attributes in the relation, it means (title, author) is a superkey for the relation and it complies with BCNF.

Since all FD's for this relation satisfy the conditions for BCNF, this table is already in BCNF. Since the table is already in BCNF and no decomposition was done, all dependencies are preserved.

order

1st FD...
Closure of *order_id*, (*order_id*)+ = (order_id, email, shipping_id, date, street_number, street_name, postal_code, city, province)
Since the closure of order_id includes all attributes in the relation, it means order_id is a superkey for the

relation and it complies with BCNF.

2nd FD...

Closure of *shipping_id*, (*shipping_id*)+ = (order_id, email, shipping_id, date, street_number, street_name, postal_code, city, province)

Since the closure of shipping_id includes all attributes in the relation, it means shipping_id is a superkey for the relation and it complies with BCNF.

3rd FD...

Closure of *postal_code*, (*postal_code*)+ = (postal_code, city, province)

Since the closure of postal_code does not include all attributes in the original relation, it means postal_code is not a superkey for the relation and thus violates the conditions for BCNF. We will need to decompose this relation.

Decomposition

Decompose into two new relations, order and region_order...

order(order_id, email, shipping id, date, street_number, street_name)

region_order(postal_code, city, province)

Now none of the functional dependencies violates BCNF since postal_code is now a superkey for the relation called *region_order*.

Dependency preservation

1. Check FD: order_id → email, shipping_id, date, street_number, street_name, postal_code, city, province
   Start with result r = order_id

   R_1 =(order_id, email, shipping_id, date, street_number, street_name, postal_code)
   $t = (result \cap R_1) + \cap R_1$
   t = (order_id, email, shipping_id, date, street_number, street_name, postal_code)
   result = (order_id) ∪ (order_id, email, shipping_id, date, street_number, street_name, postal_code)
   result = (order_id, email, shipping_id, date, street_number, street_name, postal_code)

   R_2 = (postal_code, city, province)
   $t = (result \cap R_2) + \cap R_2$
   t = (postal_code, province, city)
   result = (order_id, email, shipping_id, date, street_number, street_name, postal_code) ∪ (postal_code, province, city)
   result = (order_id, email, shipping_id, date, street_number, street_name, postal_code)

   Since result contains everything on the RHS of this FD, this dependency is preserved.

2. Check FD: shipping_id → email, order_id, date, street_number, street_name, postal_code, city, province
   Start with result r = shipping_id

   R_1 =(order_id, email, shipping_id, date, street_number, street_name, postal_code)
   $t = (result \cap R_1) + \cap R_1$
   t = (order_id, email, shipping_id, date, street_number, street_name, postal_code)
   result = (shipping_id) ∪ (order_id, email, shipping_id, date, street_number, street_name, postal_code)
   result = (order_id, email, shipping_id, date, street_number, street_name, postal_code)

   R_2 = (postal_code, city, province)
   $t = (result \cap R_2) + \cap R_2$
   t = (postal_code, province, city)
   result = (order_id, email, shipping_id, date, street_number, street_name, postal_code) ∪ (postal_code, province, city)

result = (order_id, email, shipping_id, date, street_number, street_name, postal_code)

Since result contains everything on the RHS of this FD, this dependency is preserved.

3. Check FD: postal_code → city, province
   Start with result r = postal_code

   R_1 =(order_id, email, shipping_id, date, street_number, street_name, postal_code)
   $t = (result \cap R_1) + \cap R_1$
   t = (postal_code)
   result = (postal_code) ∪ (postal_code)
   result = (postal_code)

   R_2 = (postal_code, city, province)
   $t = (result \cap R_2) + \cap R_2$
   t = (postal_code, province, city)
   result = (postal_code) ∪ (postal_code, province, city)
   result = (postal_code, province, city)

   Since result contains everything on the RHS of this FD, this dependency is preserved.

All three dependencies were shown to be preserved after running the dependency preservation algorithm, therefore the decomposition into BCNF for new relations book and region_order is dependency preserving.

books_in_order

1st FD...
Closure of *isbn, order_id*, (*isbn, order_id*)+ = (isbn, order_id, amount)
Since the closure of isbn, order_id includes all attributes in the relation, it means isbn, order_id is a superkey for the relation and it complies with BCNF.

books_in_order

1st FD...
Closure of *isbn, order_id*, (*isbn, order_id*)+ = (isbn, order_id, amount)
Since the closure of isbn, order_id includes all attributes in the relation, it means isbn, order_id is a superkey for the relation and it complies with BCNF.

customer

1st FD...
Closure of *email, password*, (*email, password*)+ = (email, password, name, phone, street_number, street_name, postal_code, city, province, card_number, admin)
Since the closure of email, password includes all attributes in the relation, it means email, password is a superkey for the relation and it complies with BCNF.

2nd FD...
Closure of *postal_code*, (*postal_code*)+ = (postal_code, city, province)
Since the closure of postal_code does not include all attributes in the original relation, it means postal_code is not a superkey for the relation and thus violates the conditions for BCNF. We will need to decompose this relation.

Decomposition
Decompose into two new relations, order and region_order...
customer(email, password, name, phone, street_number, street_name, postal_code, card_number, admin)
region_customer(postal_code, city, province)

Now none of the functional dependencies violates BCNF since postal_code is now a superkey for the relation called *region_customer*.

Dependency preservation

1. Check FD: email, password → name, phone, street_number, street_name, postal_code, city, province, card_number, admin
   Start with result r = (email, password)

   R_1 =(email, password, name, phone, street_number, street_name, postal_code, card_number, admin)
   $t = (result \cap R_1) + \cap R_1$
   t = (email, password, name, phone, street_number, street_name, postal_code, card_number, admin)
   result = (email, password) ∪ (email, password, name, phone, street_number, street_name, postal_code, card_number, admin)
   result = (email, password, name, phone, street_number, street_name, postal_code, card_number, admin)

   R_2 = (postal_code, city, province)
   $t = (result \cap R_2) + \cap R_2$
   t = (postal_code, province, city)
   result = (email, password, name, phone, street_number, street_name, postal_code, card_number, admin)
   ∪ (postal_code, province, city)
   result = (email, password, name, phone, street_number, street_name, postal_code, card_number, admin)

   Since result contains everything on the RHS of this FD, this dependency is preserved.

2. Check FD: postal_code → city, province
   Start with result r = postal_code

   R_1 =(email, password, name, phone, street_number, street_name, postal_code, card_number, admin)
   $t = (result \cap R_1) + \cap R_1$
   t = (postal_code)
   result = (postal_code) ∪ (postal_code)
   result = (postal_code)

   R_2 = (postal_code, city, province)
   $t = (result \cap R_2) + \cap R_2$
   t = (postal_code, province, city)
   result = (postal_code) ∪ (postal_code, province, city)
   result = (postal_code, province, city)

   Since result contains everything on the RHS of this FD, this dependency is preserved.

Both dependencies were shown to be preserved after running the dependency preservation algorithm, therefore the decomposition into BCNF for new relations book and region_order is dependency preserving.

publisher

1st FD...
Closure of *name*, (*name*)+ = (name, phone, bank_number, email, street_number, street_name, postal_code)
Since the closure of name includes all attributes in the relation, it means name is a superkey for the relation and it complies with BCNF.

2nd FD...
Closure of *email*, (*email*)+ = (name, phone, bank_number, email, street_number, street_name, postal_code)
Since the closure of email includes all attributes in the relation, it means email is a superkey for the relation and it complies with BCNF.

Since all FD's for this relation satisfy the conditions for BCNF, this table is already in BCNF. Since the table is already in BCNF and no decomposition was done, all dependencies are preserved.

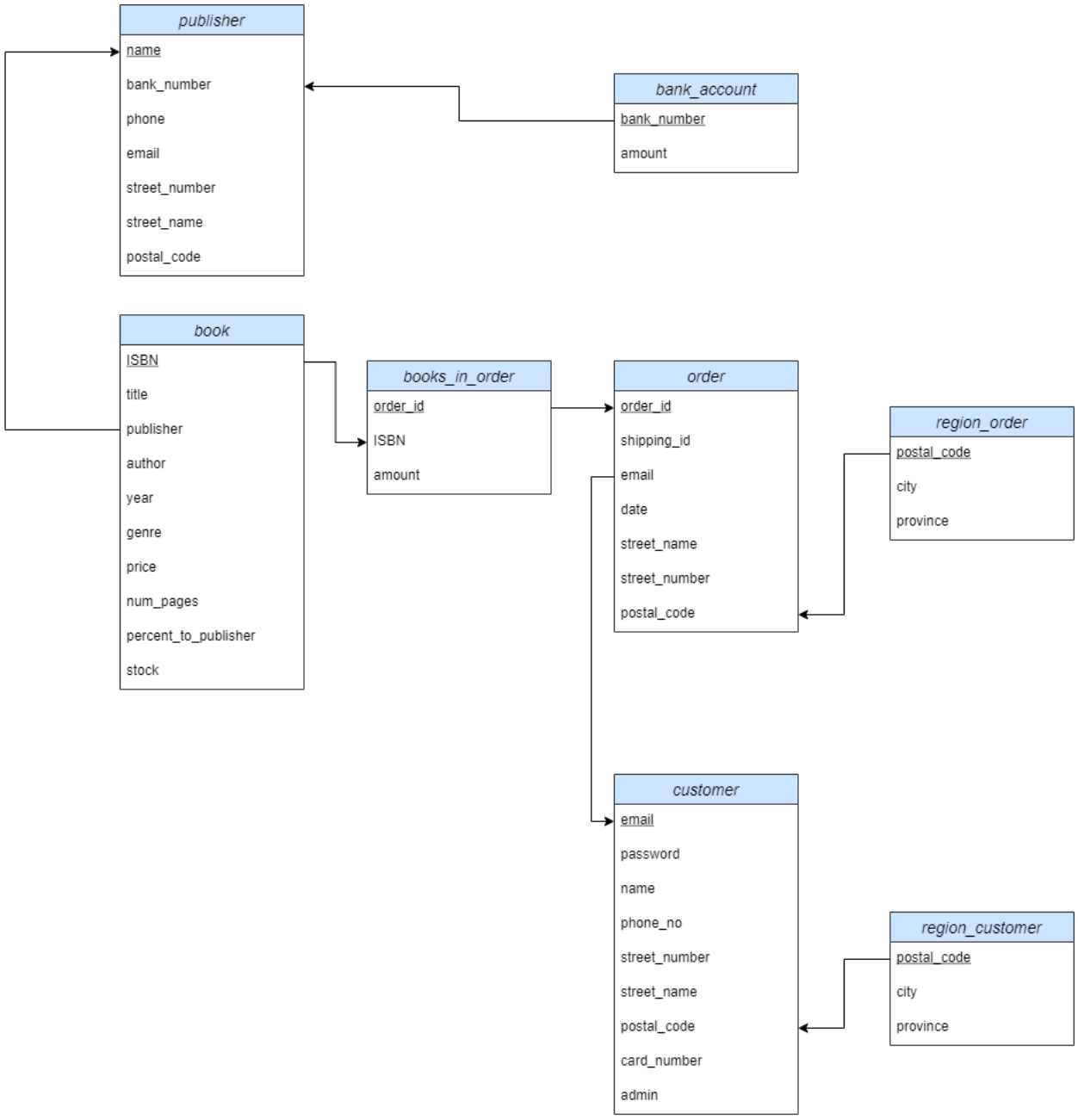
bank_account

1st FD...
Closure of *bank_number*, (*bank_number*)+ = (bank_number, amount)
Since the closure of bank_number includes all attributes in the relation, it means bank_number is a superkey for the relation and it complies with BCNF.

Since all FD's for this relation satisfy the conditions for BCNF, this table is already in BCNF. Since the table is already in BCNF and no decomposition was done, all dependencies are preserved.

4 Database Schema Diagram

The following is the Schema Diagram created to model schemas gained from our ER diagram and after normalization.


publisher
name
bank_number
phone
email
street_number
street_name
postal_code
bank_account
bank_number
amount
book
ISBN
title
publisher
author
year
genre
price
num_pages
percent_to_publisher
stock
books_in_order
order_id
ISBN
amount
order
order_id
shipping_id
email
date
street_name
street_number
postal_code
region_order
postal_code
city
province
customer
email
password
name
phone_no
street_number
street_name
postal_code
card_number
admin
region_customer
postal_code
city
province