4. Calculate total

↳ make
↳ Model
↳ Year
↳ Weight
} Total no. of attributes for vechile is (4)

↳ make
↳ Model
↳ Year
↳ No. of door
↳ Weight
↳ truck-size
} Total no. of attributes for car is (6)

↳ make
↳ Model
↳ Year
↳ Weight
↳ type of handle
↳ frame material
} Total no. of attributes is (6)

↳ make
↳ Model
↳ Year
↳ Weight
↳ Large-capacity
↳ No. of Axles
} Total no. of attributes for truck: 6

b.

The independence axiom states that the design should maintain the independence of components meaning each subclass should have attributes specific to it and not be influenced by others subclass.

Contain common attributes Bike - Type of handle bar and frame material. Car - no. of doors & trunksize specific Truck - Large capacity and no. of class specific. Each subclass has its own specific attributes in addition of inheriting common attributes from vechile class.

c.)

Vechile : 4 attributes Bike : 6 attributes (4+2) total 22 attributes.
Car : 6 attributes Truck : 6 Attributes (4+2)

Suggestion to minimize the content

To adhere to the information the information axiom content should be minimized while maintaing the designs functionality. Given the attributes are well-distributed

5. a)

$$Z = \frac{X - \mu}{\sigma}$$ Task A: $Z_A = \frac{6-5}{1} = 1$, Task B: $\frac{12-10}{2} = 1$, Task C: $\frac{25-20}{5}$

b.)

Task C has the highest z-score.

Practical

↳ Task C is higher deviation suggest what the actual users take to complete this is significantly longer than expected.
↳ The increased time may result is user frustration or error, suggesting need for redesign or better user guidance for task C.

z score for less than 5 mins:

$z = \frac{5-6}{1} = -1$

Probability ≈ 0.1587

z score for less than 10 mins

$z = \frac{10-12}{2} = -1$

Probability ≈ 0.1587

z-score for less than 20 mins:

$z = \frac{20-25}{3} = -1.67$

Probability = 0.0475

To estimate the total effort required to compare complete the development and quality associative phase of the project, we need to break down the problem

1. Calculate total no of methods
   Total Methods = 5 × 4 = 20
2. Calculate total lines of code
   LOC = 20 × 25 = 500
3. Estimate development:
   Effort = Total LOC × 14
   = 700 person-hours
4. Estimate quality assurance effort
   = 0.20 × 70
   = 140 person hours
5. Calculate total effort
   = 700 + 140
   = 840 person-hours

Components:

1) Client Interface
2) Transaction manager
3) Account db
4) Authentication Service
5) Notification Service.

1) Client Interface
2) Transaction manager
3) Account db
4) Authentication Service
5) Notification service.

1) ClientInterface → Transaction Manager
2) Transaction Manager → Account db
3) Transaction manager → Authentication Service
4) Transaction manager → Notification Service.

Client Server (Hosts client interface and Transaction Manager)
Database Server (Hosts account db)
Authentify Server (Hosts Authentication)

1) Client Server → Db Server
2) Client Server → Authentify Server.

Db Server: Hosts client Interface and transaction manager Compone
Hosts account db component.
Hosts the Authentication service And Notification Service.

8.) a)

1.) Library - Book (A library consists many books)
2.) Library - Member (A library consists many member)
3.) Library - Loan (A library manage many loans.)
4.) Book Loan (A book can be loaded multiple times)
5.) Member Loan (A member loan can have multiple loans)

b.)

1.) Library - Book Library : 1 : Book : 0
2.) Library - Member Library : 1 Member : 0
3.) Library - Loan Library : 1 Loan : 0
4.) Book - Loan Book : 1 Loan : 0
5.) Member Loan : Member : 1 Loan : 0

c)

1.) Library and Book: Composition
The library own books and responsible for its life cycle
2.) Library and Member: Aggregation
The library holds members but members independence of library.
3.) Library and Loan: Aggregation
The library manage loan but loans independence of library life cycle
4.) Book and Loan: Association
A book can be learned multiple times and records simple reference
5.) Member Loan: Association.
A member can have multiple loans.

9.)

Product Table:

- ProductId (bigint, primary key)
- Name (String)
- Description (String)
- Price (Float)
- Quantity Stock (int)

Customer Table:

- CustomerID (int, primary key)
- First Name (String)
- Last name (String)
- Email (String)
- Phone number (String)

Order Table:

- OrderId (int, primary key)
- OrderDate (date)
- CustomerId (int)
- Total Amount (float)

4.) Order Item Table:

- OrderItemId (primary key int)
- OrderId (int, forgin key)
- ProductId (int)
- Quantity : (int)
- Itemprice (int)

b.)

1.) Product Table: 5 2.) Customer Table: 5 3.) Order Table : 4
4.) Order Item : 5 Total = 19 Attributes.

c)

1.) Product Table : 200 products 2.) Customer Table : 150 customers 3.)
4.) Order Item Table : Total: 4500 * 5 = 7760. Total : 7500

10.

Given: Total no. of classes : 500 Avg no. of methods per class : 10
Total no. of test method : no. of classes x Avg no. of methods per class = 5000

Given : 85% method coverage. Test case per method to achieved 85% coverage
Total no. of test case = Total no. of methods x Test case per method = 50

Given ; 0.5 defects per method.
Expected no of defects : Total no. of Methods x Defects per method = 2500

Given ; Defect detection efficiency (DDE) = 70%.
Effected no. of defects and fixed. = Expected no of defects x DDE
= 1750

Given ; Expected no. of defects before testing expected number of defects detected and fixed
= 2500 - 1750 = 750.