Usage of Design Tools like Argo UML

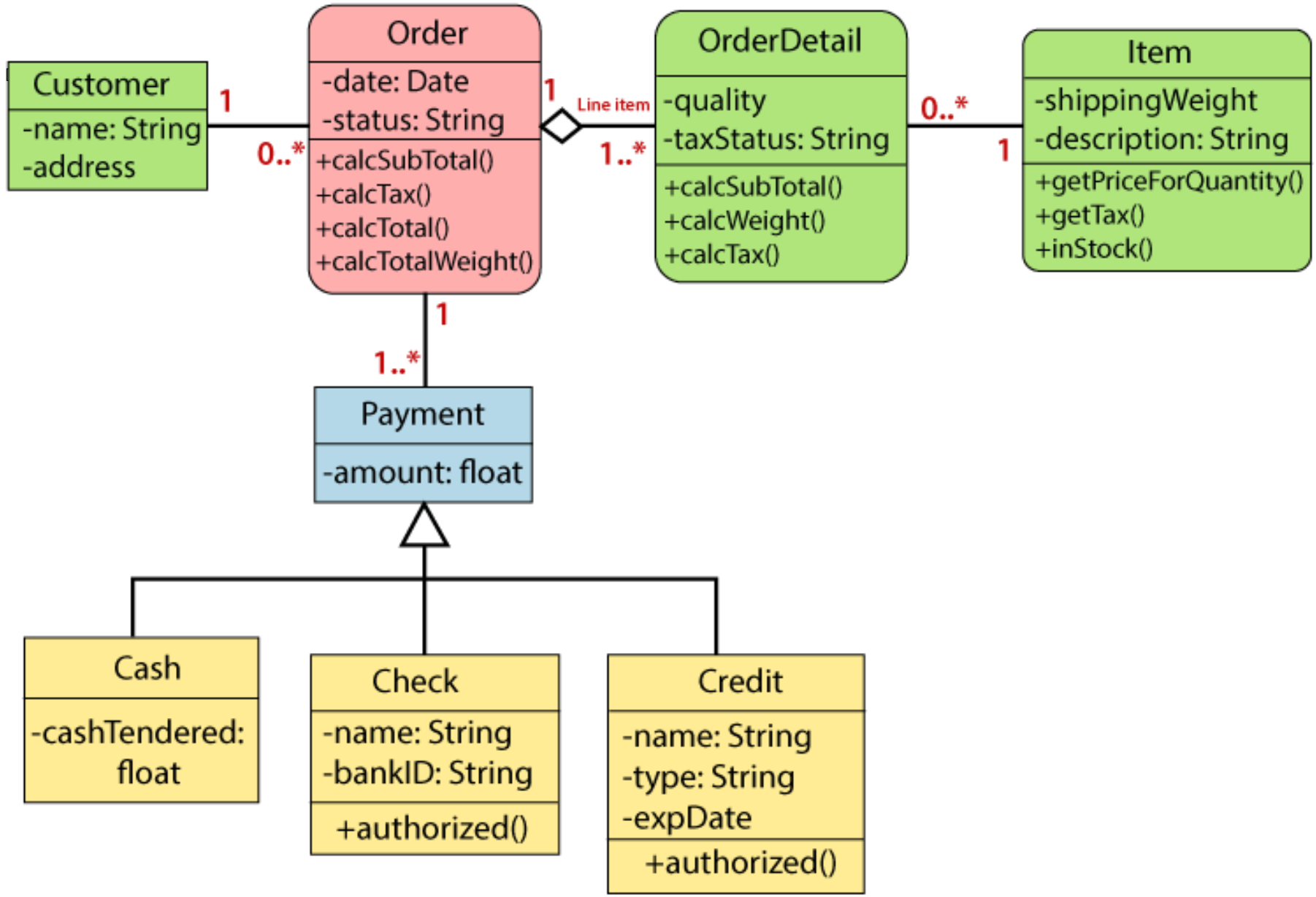
ArgoUML

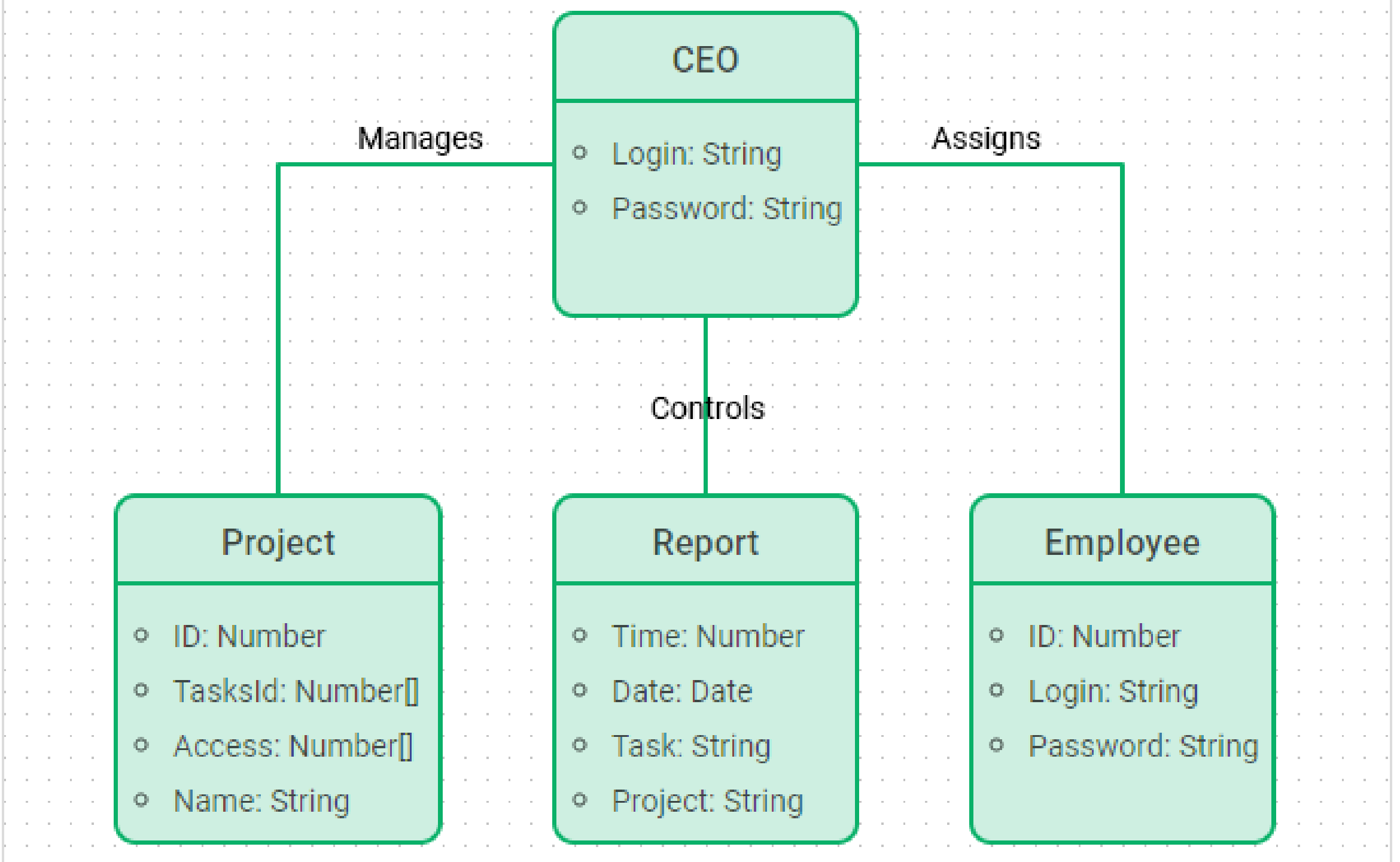
- ArgoUML is an UML diagramming application written in Java and released under the open source Eclipse Public License. By virtue of being a Java application, it is available on any platform supported by Java SE.
- Argo UML is a popular design tool used for creating Unified Modeling Language (UML) diagrams. UML is a visual modeling language used for software development, and it is used to describe the structure and behavior of software systems.
- UML provides a graphical user interface for creating UML diagrams and models.

Here are some of the common use cases and scenarios where Argo UML can be useful:

- **Creating class diagrams**: Argo UML can be used to create class diagrams, which are used to describe the structure of software systems. Class diagrams show the classes, interfaces, attributes, and methods of a system, and the relationships between them.

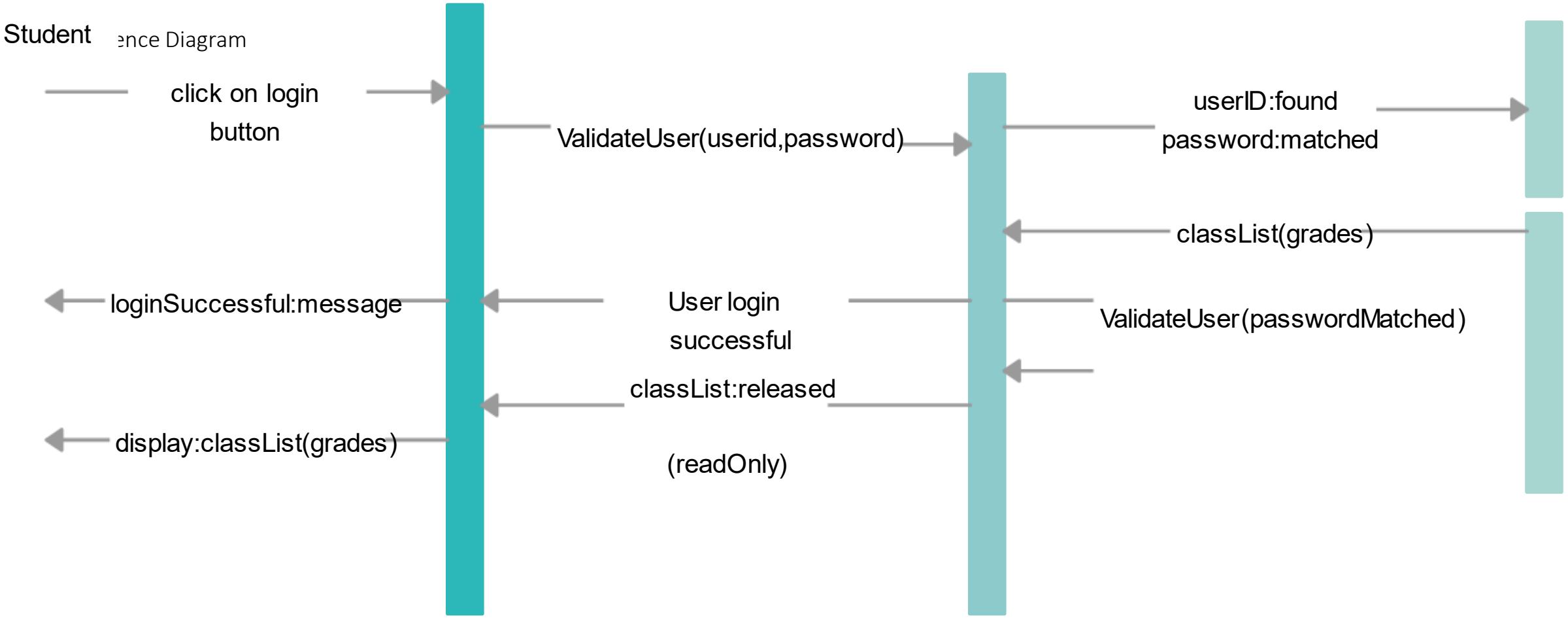
UML CLASS


Customer
-name: String
-address
1
0..*
Order
-date: Date
-status: String
+calcSubTotal()
+calcTax()
+calcTotal()
+calcTotalWeight()
1
Line item
1..*
OrderDetail
-quality
-taxStatus: String
+calcSubTotal()
+calcWeight()
+calcTax()
0..*
1
Item
-shippingWeight
-description: String
+getPriceForQuantity()
+getTax()
+inStock()
1
1..*
Payment
-amount: float
Cash
-cashTendered: float
Check
-name: String
-bankID: String
+authorized()
Credit
-name: String
-type: String
-expDate
+authorized()


CEO
Login: String
Password: String
Manages
Assigns
Controls
Project
ID: Number
TasksId: Number[]
Access: Number[]
Name: String
Report
Time: Number
Date: Date
Task: String
Project: String
Employee
ID: Number
Login: String
Password: String

SEQUENCE DIAGRAM

- Argo UML can be used to create sequence diagrams, which are used to describe the interactions between objects in a software system. Sequence diagrams show the messages exchanged between objects and the order in which they are exchanged.


Student
ence Diagram
click on login button
ValidateUser(userid,password)
userID:found
password:matched
classList(grades)
loginSuccessful:message
User login successful
ValidateUser(passwordMatched)
classList:released
display:classList(grades)
(readOnly)

System

Actor

Log in ()

Authorization()

BrowsEvents()

Display Events

Request Event Details

Show Event Details

Registration

Confirmation

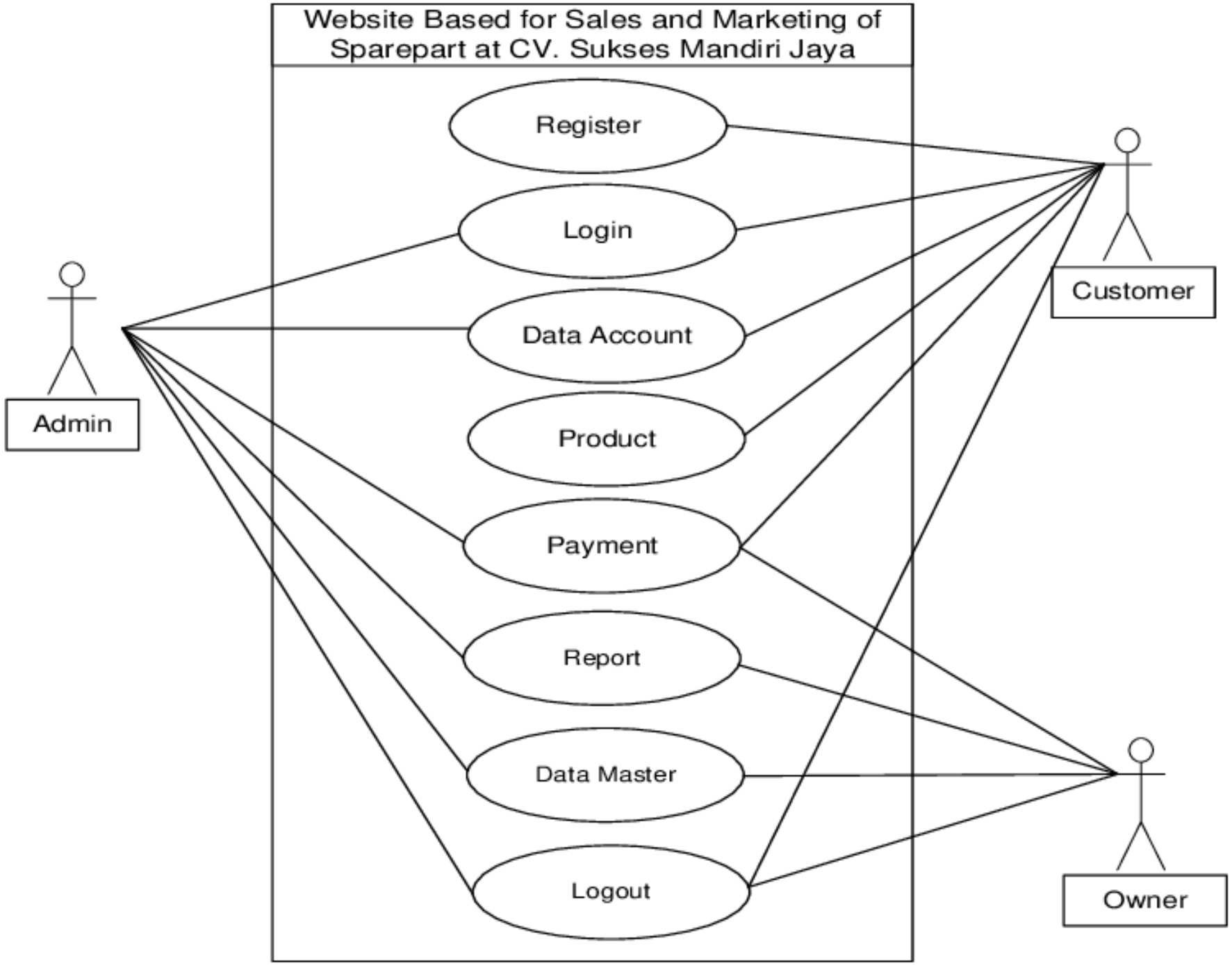
Proceed to check-out

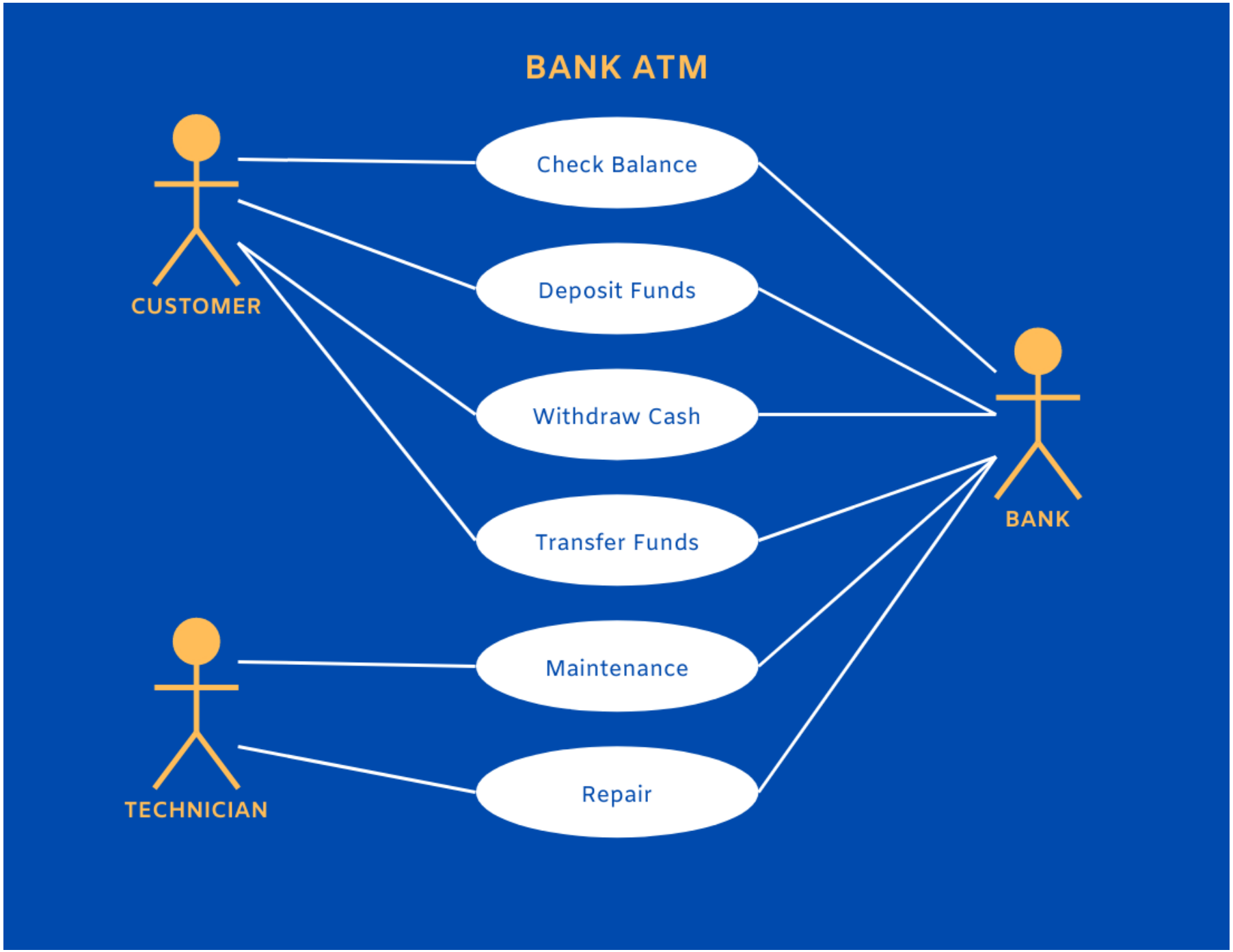
Billing Information

Payment

USE CASE DIAGRAM

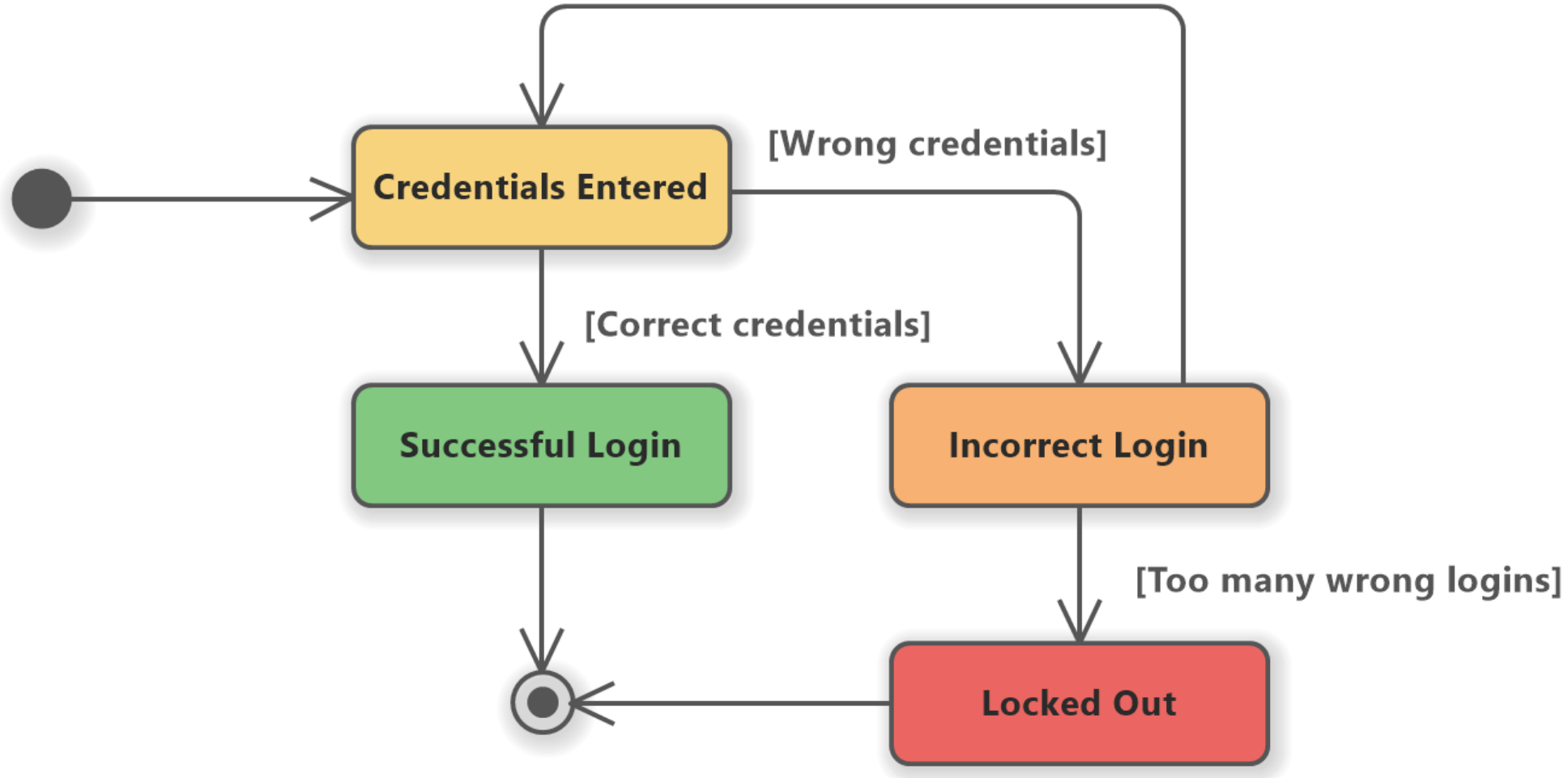
- Argo UML can be used to create use case diagrams, which are used to describe the behavior of software systems from the perspective of its users. Use case diagrams show the actors, use cases, and relationships between them.


Website Based for Sales and Marketing of Sparepart at CV. Sukses Mandiri Jaya
Register
Login
Data Account
Product
Payment
Report
Data Master
Logout
Admin
Customer
Owner

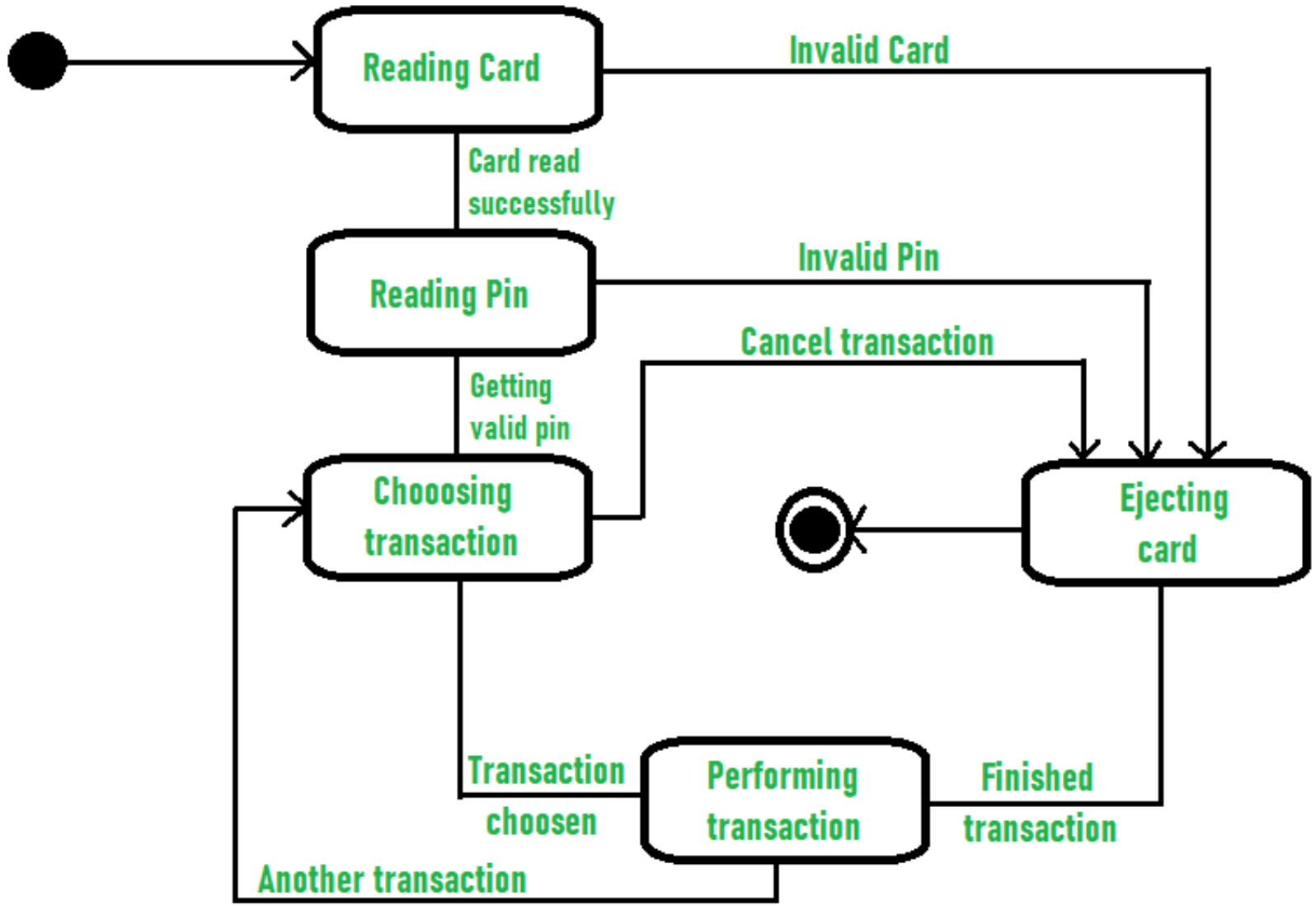

BANK ATM
CUSTOMER
Check Balance
Deposit Funds
Withdraw Cash
Transfer Funds
BANK
TECHNICIAN
Maintenance
Repair

STATE MACHINE DIAGRAM

- Argo UML can be used to create state machine diagrams, which are used to describe the behavior of objects in a software system. State machine diagrams show the states, events, and transitions of objects and the conditions under which transitions occur



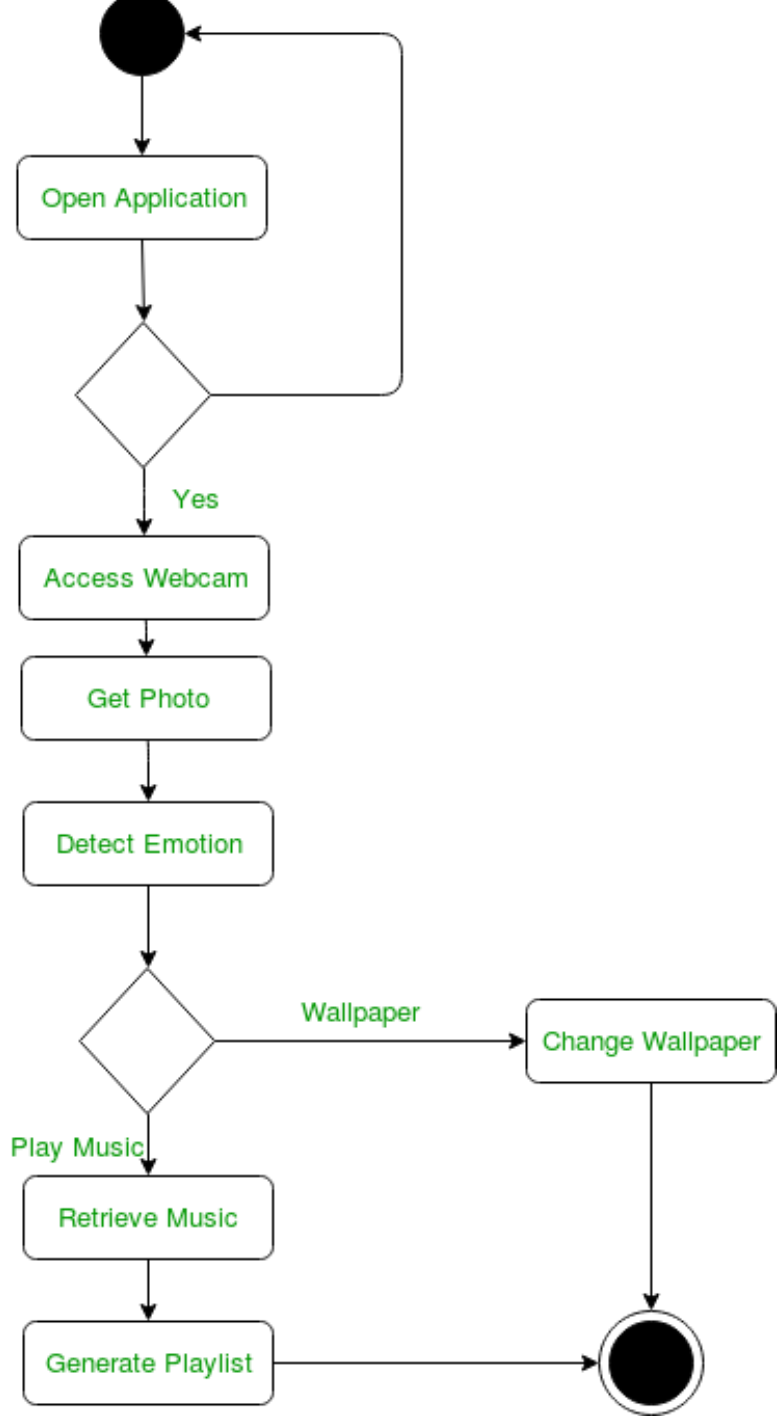
SOFTWARE IDEAS MODELER

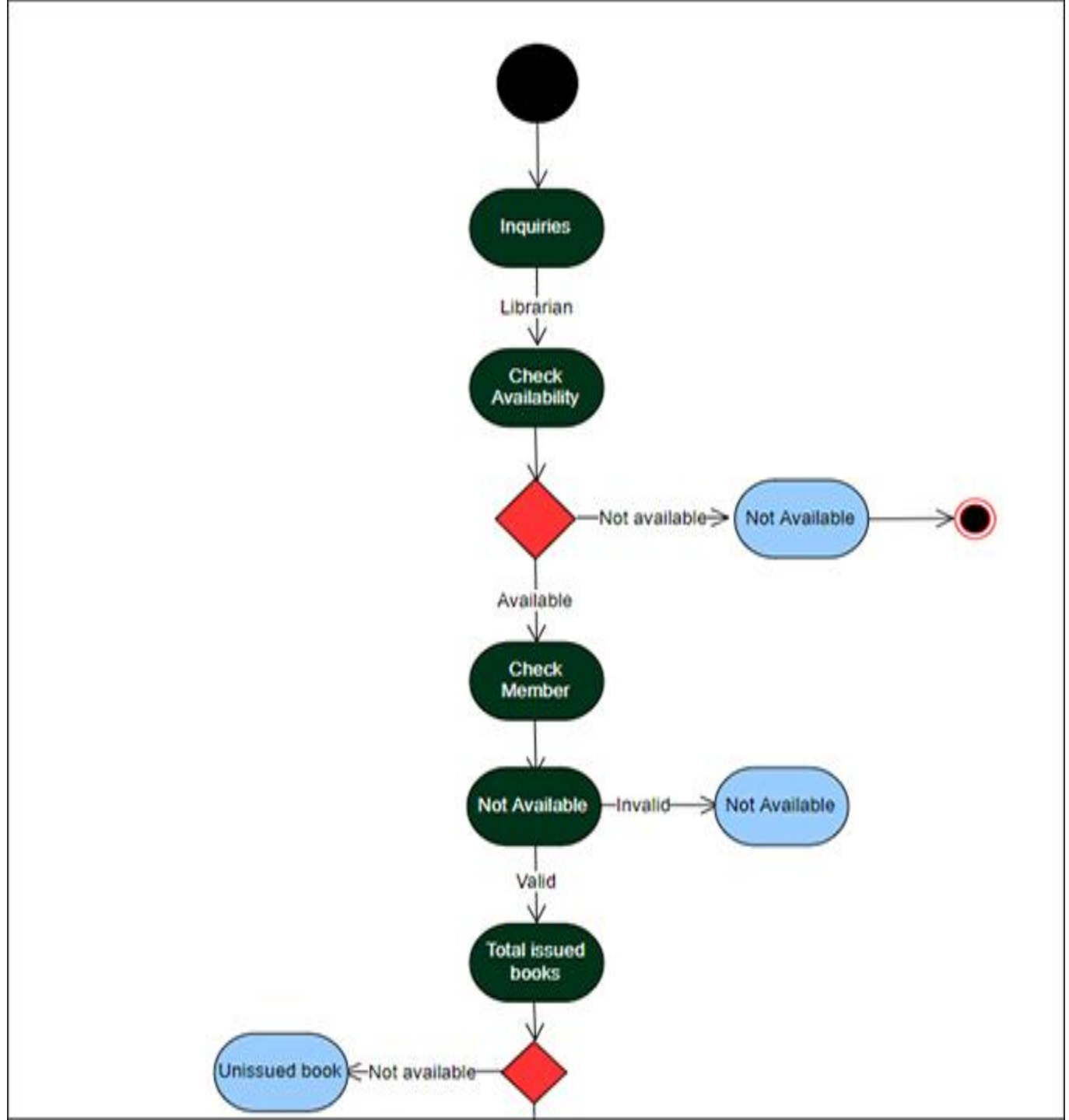


State Transition Diagram for ATM System

ACTIVITY DIAGRAM

- Argo UML can be used to create activity diagrams, which are used to describe the flow of activities in a software system. Activity diagrams show the actions, control flows, and object flows of a system.


Open Application
Yes
Access Webcam
Get Photo
Detect Emotion
Wallpaper
Change Wallpaper
Play Music
Retrieve Music
Generate Playlist


Inquiries
Librarian
Check Availability
Not available
Not Available
Available
Check Member
Not Available
Invalid
Not Available
Valid
Total issued books
Unissued book
Not available

1. What is Argo UML used for?

a) Creating UML diagrams

b) Generating code

c) Testing software

d) None of the above

2. What is a UML diagram?

a) A diagram used for testing software

b) A diagram used for generating code

c) A visual representation of object-oriented software systems

d) None of the above

3. What types of UML diagrams can be created using Argo UML?

a) Use case diagrams

b) Class diagrams

c) Sequence diagrams

d) All of the above

4. What is the purpose of annotating UML diagrams?

a) To add color to the diagrams

b) To clarify ideas and capture important information

c) To reduce the size of the diagrams

d) None of the above

5. Can Argo UML automatically generate documentation based on UML diagrams?

a) Yes

b) No

How can documentation generated by Argo UML be used?

a) To prepare detailed notes about a software system

b) To create software prototypes

c) To debug software

d) None of the above

Can Argo UML be used to collaborate with others on UML diagrams?

a) Yes

b) No

What is the main advantage of using UML diagrams for software design?

a) They provide a graphical representation of the software system

b) They automate the software development process

c) They allow for easier debugging of software

d) None of the above

What is the purpose of a use case diagram?

a) To describe the behavior of a software system

b) To describe the structure of a software system

c) To describe the relationships between objects in a software system

d) None of the above

What is the purpose of a class diagram?

a) To describe the behavior of a software system

b) To describe the structure of a software system

c) To describe the relationships between objects in a software system

d) None of the above

What is the purpose of a sequence diagram?

a) To describe the behavior of a software system

b) To describe the structure of a software system

c) To describe the relationships between objects in a software system

d) None of the above

What is the purpose of an activity diagram?

a) To describe the behavior of a software system

b) To describe the structure of a software system

c) To describe the relationships between objects in a software system

d) None of the above

Create a class diagram using Argo UML for a simple banking system. The system should include classes for customers, accounts, and transactions. Each customer can have multiple accounts, and each account can have multiple transactions.

Answers:

The resulting class diagram should show three classes (Customer, Account, and Transaction) and their attributes and methods.

The "Customer" class should have attributes such as "Name", "Address", and "Phone Number", and methods such as "Create Account", "Close Account", and "View Transactions".

The "Account" class should have attributes such as "Account Number", "Balance", and "Type", and methods such as "Deposit", "Withdraw", and "View Transactions".

The "Transaction" class should have attributes such as "Transaction ID", "Date", and "Amount", and methods such as "View Details" and "Print Receipt". arrows to show the direction of the associations.

The "Customer" class should have an association with the "Account" class with a multiplicity of "1 to many", indicating that a customer can have multiple accounts. The "Account" class should have an association with the "Transaction" class with a multiplicity of "1 to many", indicating that an account can have multiple transactions. The class diagram should have clear labels and