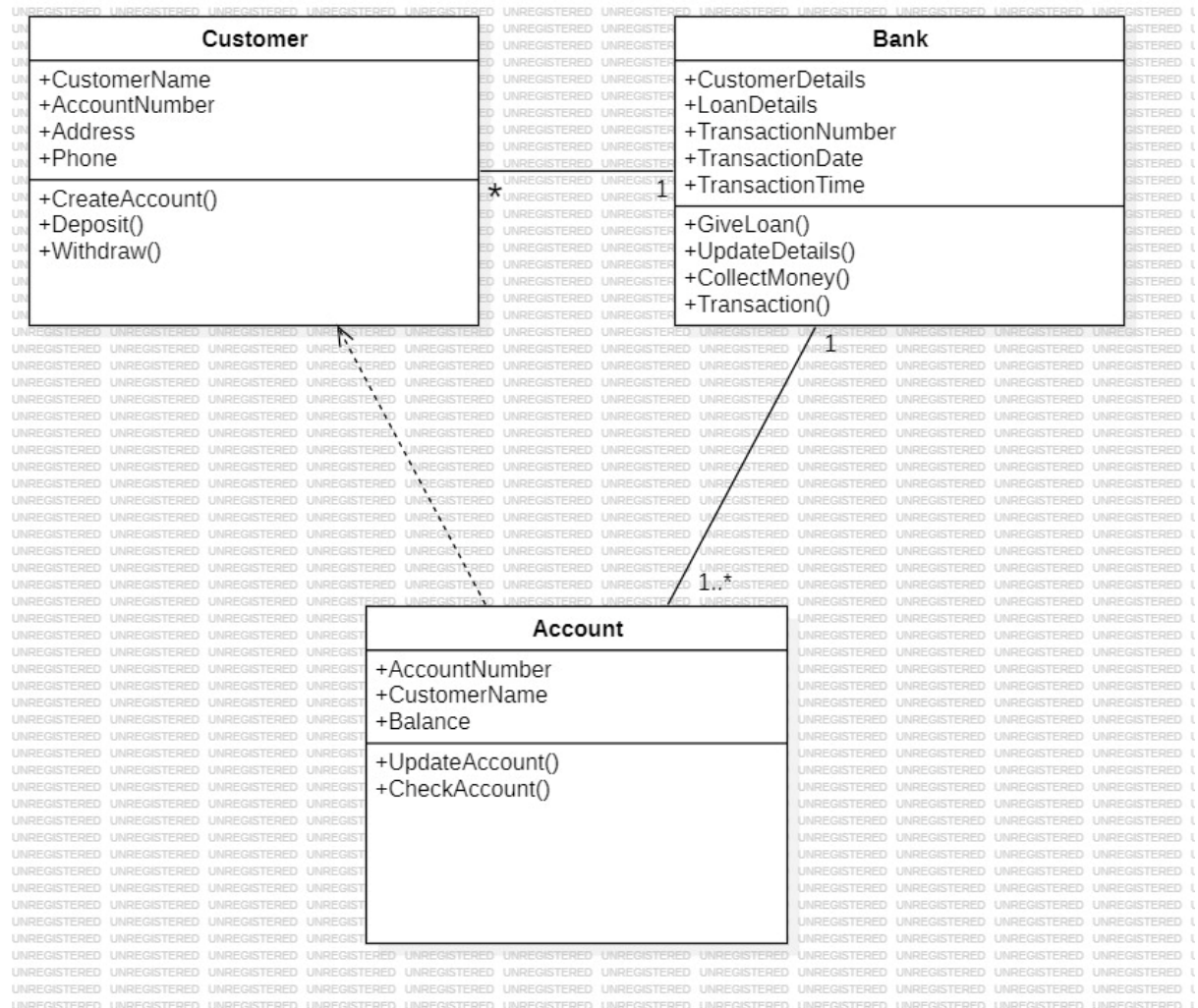


UML DIAGRAMS

G.Mahidhar – BU22CSEN0101757

Class Diagram :



CODE :

class Customer:

```
def __init__(self, customer_name, account_number, address, phone):
```

```
    self.customer_name = customer_name
```

```
    self.account_number = account_number
```

```
    self.address = address
```

```
    self.phone = phone
```

```
def create_account(self):
```

```
    # Logic to create an account
```

```

        pass

    def deposit(self, amount):
        # Logic to deposit money
        pass

    def withdraw(self, amount):
        # Logic to withdraw money
        pass

class Account:
    def __init__(self, account_number, customer_name, balance=0):
        self.account_number = account_number
        self.customer_name = customer_name
        self.balance = balance

    def update_account(self):
        # Logic to update account details
        pass

    def check_account(self):
        return self.balance

class Bank:
    def __init__(self, customer_details):
        self.customer_details = customer_details # A list of customers
        self.loan_details = []

    def give_loan(self, customer, amount):
        # Logic to give a loan to a customer
        pass

    def update_details(self):
        # Logic to update bank details
        pass

    def collect_money(self, customer, amount):
        # Logic to collect money
        pass

```

```

def transaction(self):

    # Logic to handle transactions

    pass

# Example of creating objects and using the classes

if __name__ == "__main__":

    # Creating a customer

    customer1 = Customer("John Doe", "12345", "123 Elm St", "555-1234")

    # Creating an account for the customer

    account1 = Account(customer1.account_number, customer1.customer_name, 1000)

    # Creating a bank

    bank = Bank(customer_details=[customer1])

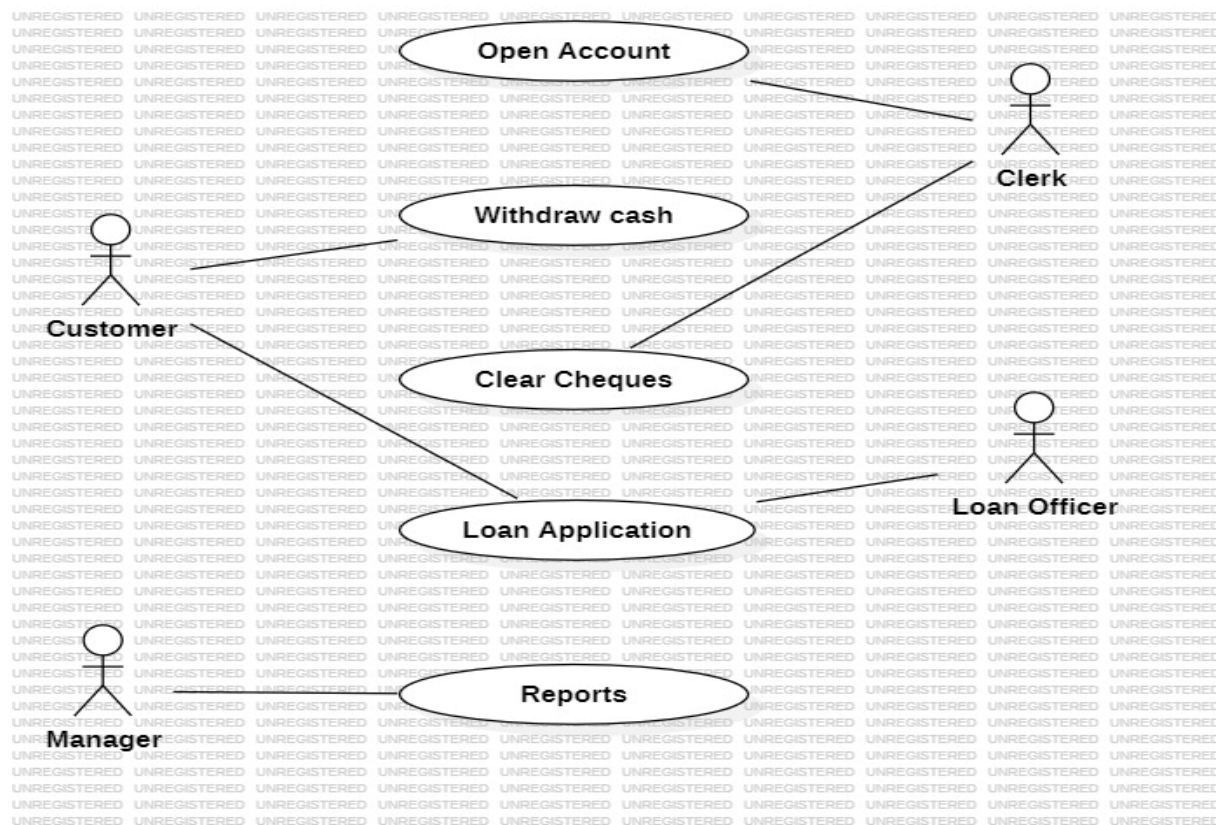
    # Example actions

    account1.deposit(500)

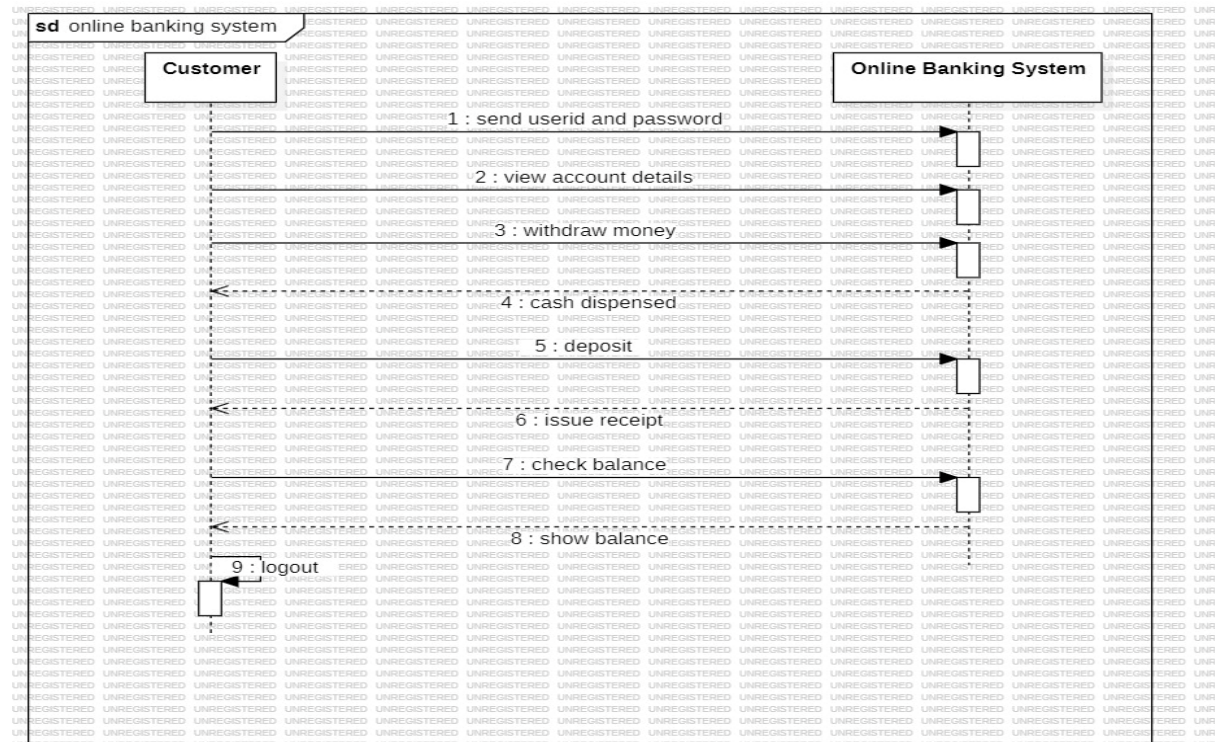
    account1.withdraw(200)

```

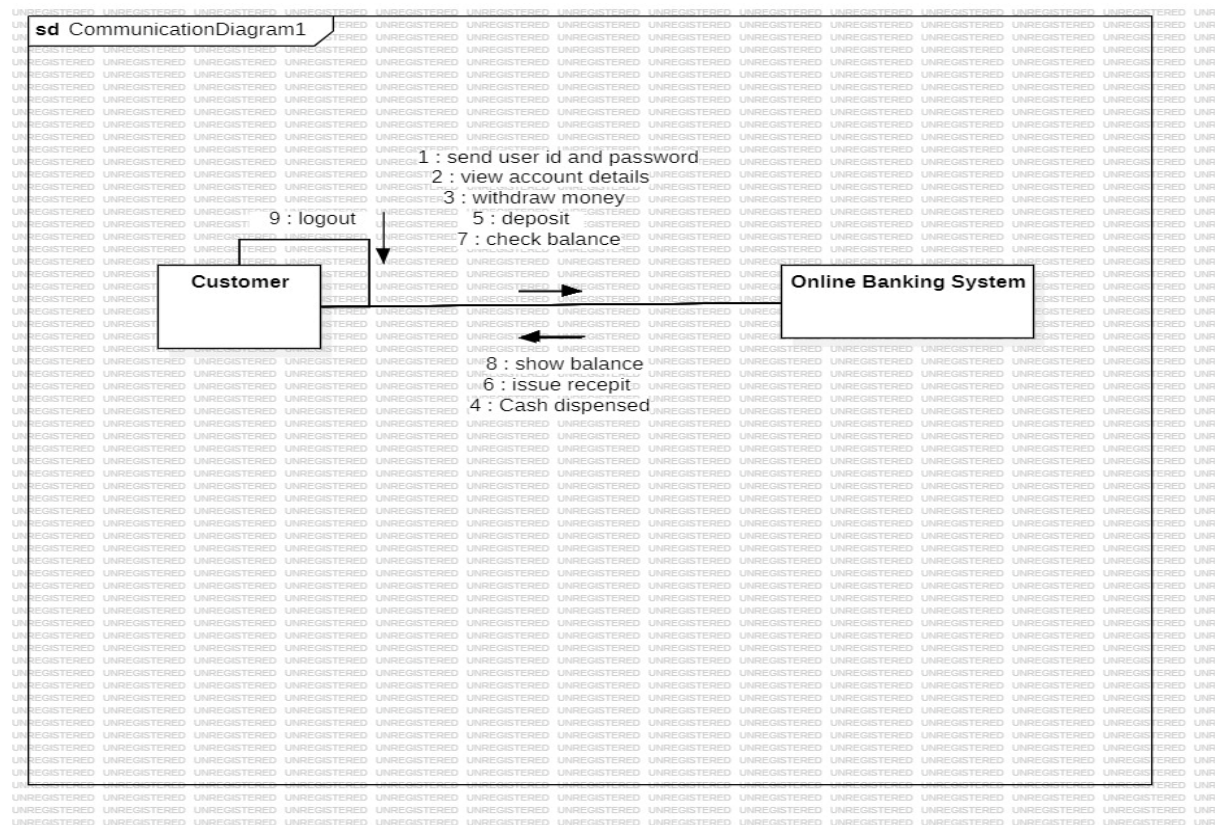
Use Case Diagram :



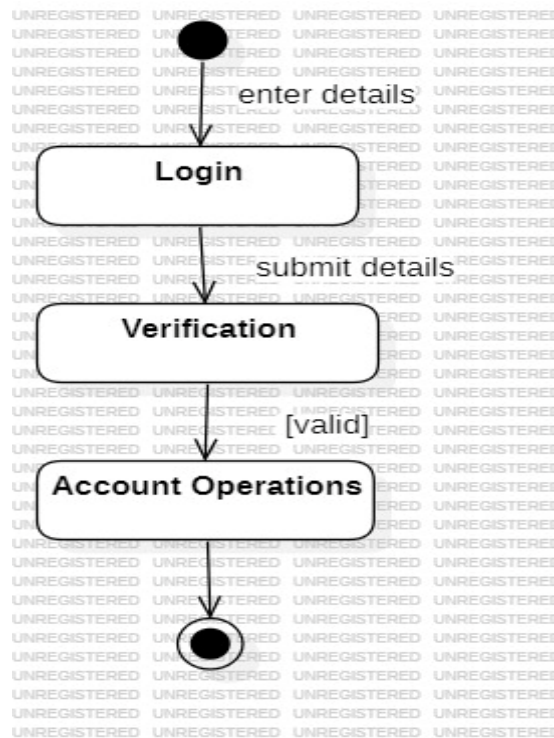
Sequence Diagram :



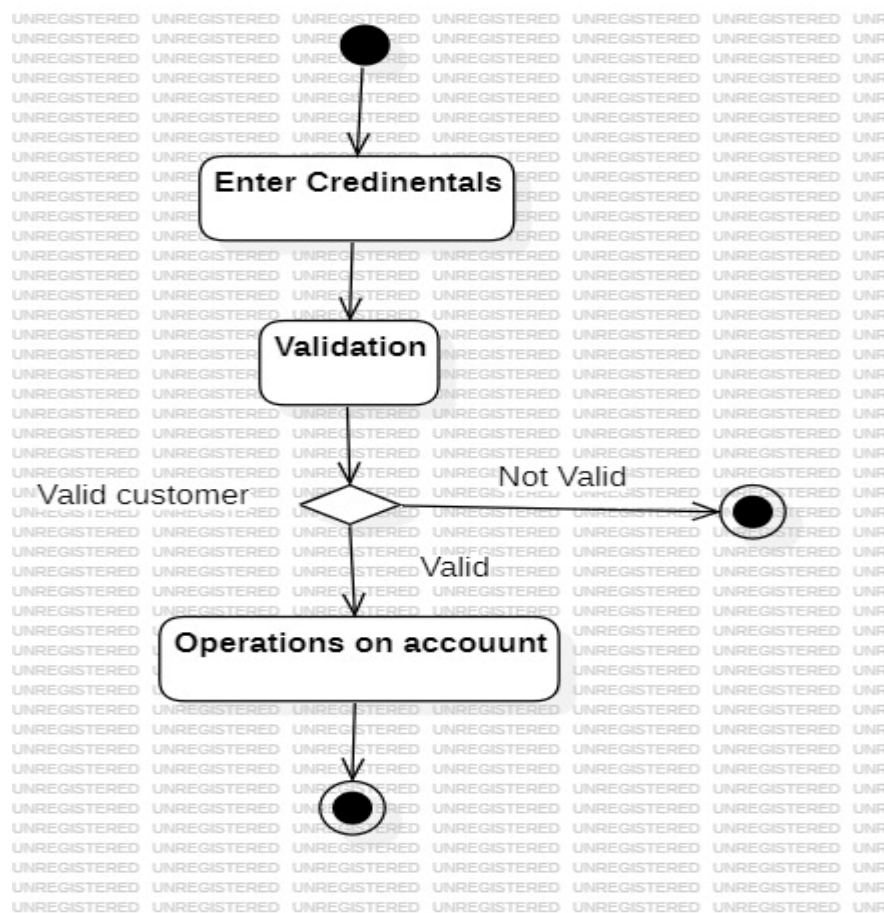
Collaboration Diagram :



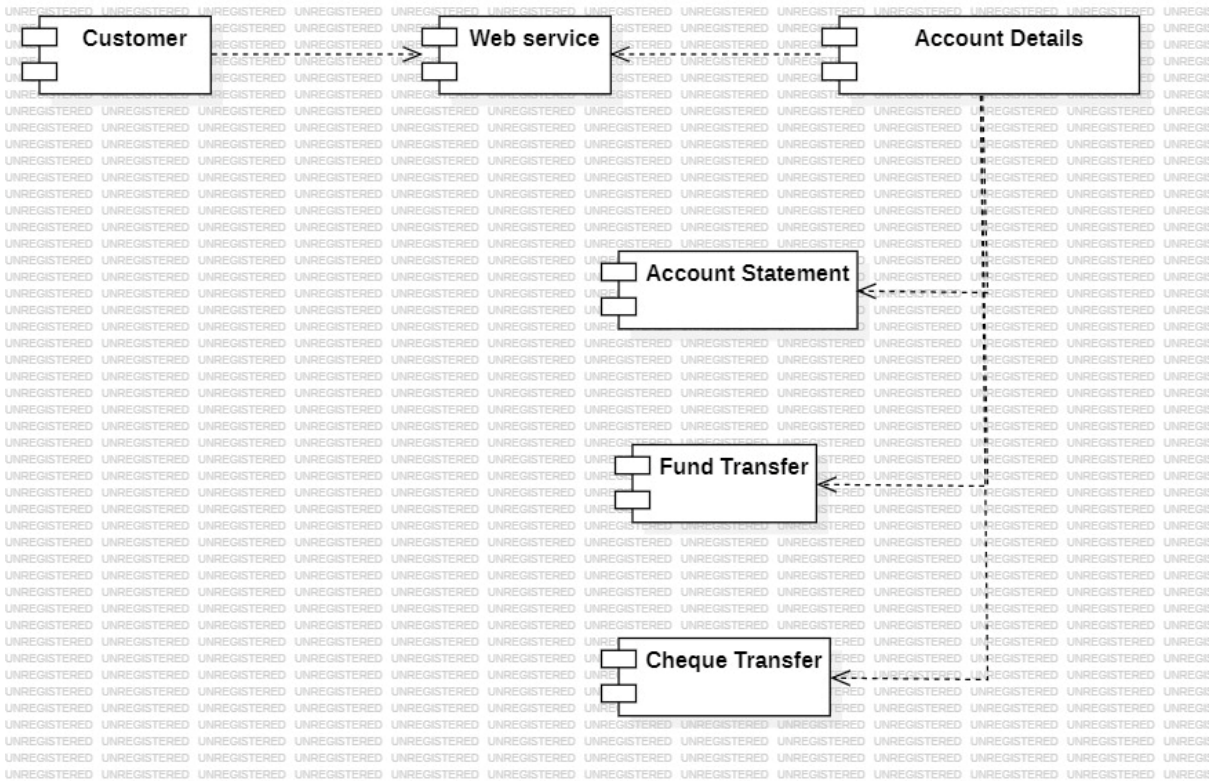
Statechart Diagram :



Activity Diagram :



Component Diagram :



Deployment Diagram :

