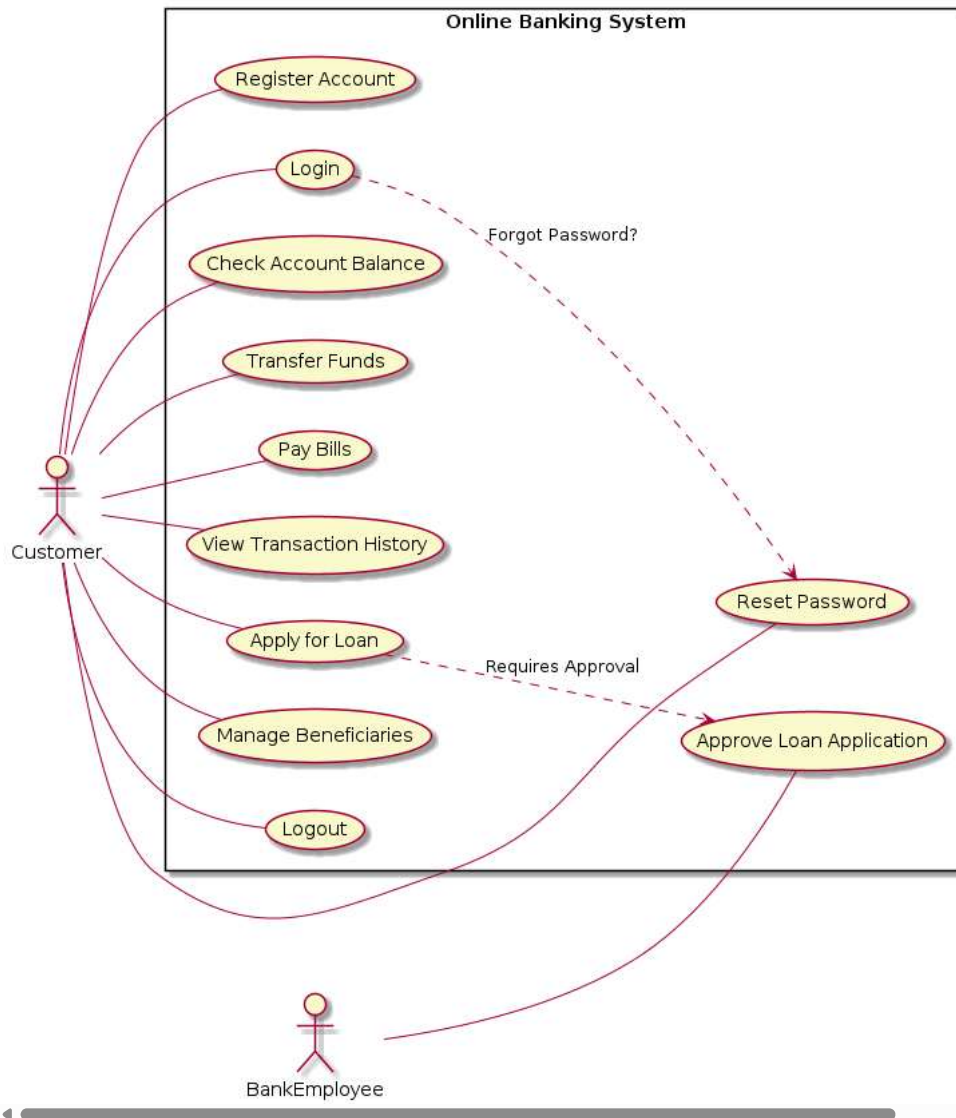


```
1 # Install PlantUML (if not already installed)
2 !apt-get install -y plantuml
3
4 # Import necessary modules
5 import os
6 from IPython.display import Image, display
7
8 # PlantUML code as a string
9 uml_code = """
10 @startuml
11 left to right direction
12 actor Customer
13 actor BankEmployee
14
15 rectangle "Online Banking System" {
16     usecase "Register Account" as UC1
17     usecase "Login" as UC2
18     usecase "Check Account Balance" as UC3
19     usecase "Transfer Funds" as UC4
20     usecase "Pay Bills" as UC5
21     usecase "View Transaction History" as UC6
22     usecase "Apply for Loan" as UC7
23     usecase "Manage Beneficiaries" as UC8
24     usecase "Logout" as UC9
25     usecase "Approve Loan Application" as UC10
26     usecase "Reset Password" as UC11
27 }
28 Customer -- UC1
29 Customer -- UC2
30 Customer -- UC3
31 Customer -- UC4
32 Customer -- UC5
33 Customer -- UC6
34 Customer -- UC7
35 Customer -- UC8
36 Customer -- UC9
37 Customer -- UC11
38 BankEmployee -- UC10
39 UC7 ..> UC10 : "Requires Approval"
40 UC2 ..> UC11 : "Forgot Password?"
41 @enduml
42 """
43
44 # Define file paths
45 uml_file_path = "/content/diagram.puml"
46 output_image_path = "/content/diagram.png"
47
48 # Create a temporary file for the UML code
49 with open(uml_file_path, "w") as file:
50     file.write(uml_code)
51
52 # Generate the PlantUML diagram as PNG
53 os.system(f"plantuml -tpng {uml_file_path}")
54
55 # Check if the image was generated successfully
56 if os.path.exists(output_image_path):
57     print("PlantUML diagram generated successfully!")
58     display(Image(filename=output_image_path))
59 else:
60     print("Error: Failed to generate PlantUML diagram.")
61
```

↻ Reading package lists... Done  
 Building dependency tree... Done  
 Reading state information... Done  
 plantuml is already the newest version (1:1.2020.2+ds-1).  
 0 upgraded, 0 newly installed, 0 to remove and 29 not upgraded.  
 PlantUML diagram generated successfully!

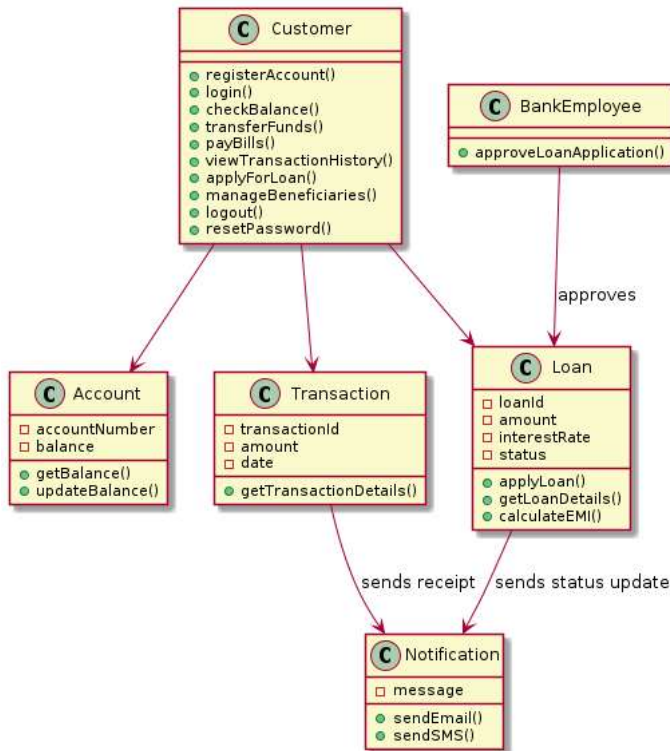


```

1 # Install PlantUML (if not already installed)
2 !apt-get install -y plantuml
3
4 # Import necessary modules
5 import os
6 from IPython.display import Image, display
7
8 # PlantUML code for the class diagram
9 uml_code = ""
10 @startuml
11 class Customer {
12   +registerAccount()
13   +login()
14   +checkBalance()
15   +transferFunds()
16   +payBills()
17   +viewTransactionHistory()
18   +applyForLoan()
19   +manageBeneficiaries()
20   +logout()
21   +resetPassword()
22 }
23
24 class BankEmployee {
25   +approveLoanApplication()
  
```

```
26 }
27
28 class Account {
29     -accountNumber
30     -balance
31     +getBalance()
32     +updateBalance()
33 }
34
35 class Transaction {
36     -transactionId
37     -amount
38     -date
39     +getTransactionDetails()
40 }
41
42 class Loan {
43     -loanId
44     -amount
45     -interestRate
46     -status
47     +applyLoan()
48     +getLoanDetails()
49     +calculateEMI()
50 }
51
52 class Notification {
53     -message
54     +sendEmail()
55     +sendSMS()
56 }
57
58 Customer --> Account
59 Customer --> Transaction
60 Customer --> Loan
61 BankEmployee --> Loan : "approves"
62 Loan --> Notification : "sends status update"
63 Transaction --> Notification : "sends receipt"
64 @enduml
65 """
66
67 # Define file paths
68 uml_file_path = "/content/diagram.puml"
69 output_image_path = "/content/diagram.png"
70
71 # Create a temporary file for the UML code
72 with open(uml_file_path, "w") as file:
73     file.write(uml_code)
74
75 # Generate the PlantUML diagram as PNG
76 os.system(f"plantuml -tpng {uml_file_path}")
77
78 # Check if the image was generated successfully
79 if os.path.exists(output_image_path):
80     print("✅ PlantUML class diagram generated successfully!")
81     display(Image(filename=output_image_path))
82 else:
83     print("❌ Error: Failed to generate the PlantUML class diagram.")
84
```

➡ Reading package lists... Done  
 Building dependency tree... Done  
 Reading state information... Done  
 plantuml is already the newest version (1:1.2020.2+ds-1).  
 0 upgraded, 0 newly installed, 0 to remove and 29 not upgraded.  
 ✔ PlantUML class diagram generated successfully!



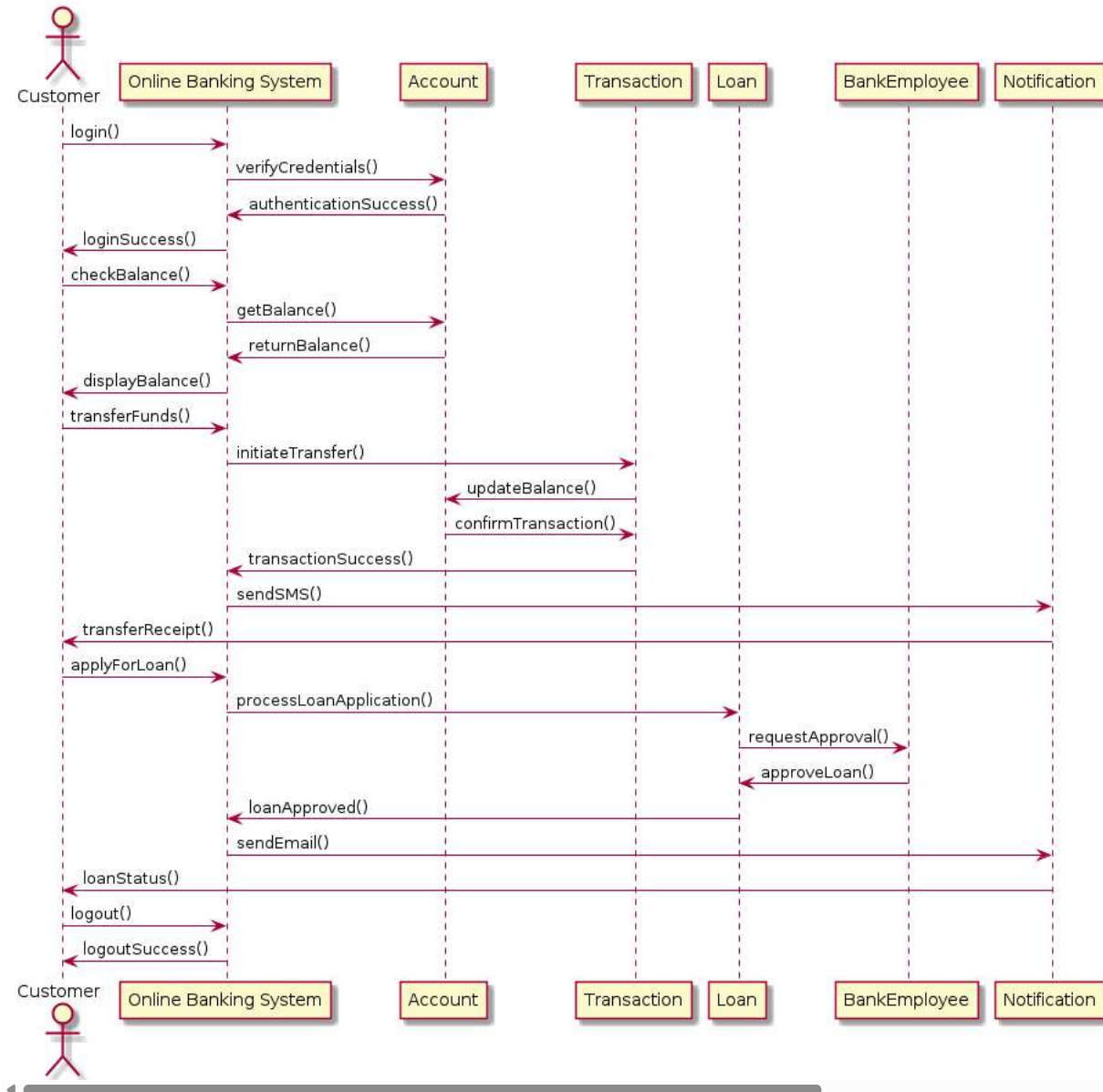
```

1 # Install PlantUML (if not already installed)
2 !apt-get install -y plantuml
3
4 # PlantUML code for the sequence diagram
5 uml_code = ""
6 @startuml
7 actor Customer
8 participant "Online Banking System" as OBS
9 participant Account
10 participant Transaction
11 participant Loan
12 participant BankEmployee
13 participant Notification
14
15 Customer -> OBS: login()
16 OBS -> Account: verifyCredentials()
17 Account -> OBS: authenticationSuccess()
18 OBS -> Customer: loginSuccess()
19
20 Customer -> OBS: checkBalance()
21 OBS -> Account: getBalance()
22 Account -> OBS: returnBalance()
23 OBS -> Customer: displayBalance()
24
25 Customer -> OBS: transferFunds()
26 OBS -> Transaction: initiateTransfer()
27 Transaction -> Account: updateBalance()
28 Account -> Transaction: confirmTransaction()
29 Transaction -> OBS: transactionSuccess()
30 OBS -> Notification: sendSMS()
31 Notification -> Customer: transferReceipt()
32
33 Customer -> OBS: applyForLoan()
34 OBS -> Loan: processLoanApplication()
35 Loan -> BankEmployee: requestApproval()
36 BankEmployee -> Loan: approveLoan()
37 Loan -> OBS: loanApproved()
38 OBS -> Notification: sendEmail()
39 Notification -> Customer: loanStatus()
  
```

```

40
41 Customer -> OBS: logout()
42 OBS -> Customer: logoutSuccess()
43 @enduml
44 """
45
46 # Create a temporary file for the UML code
47 uml_file_path = "/content/sequence_diagram.puml"
48 with open(uml_file_path, "w") as file:
49     file.write(uml_code)
50
51 # Generate the PlantUML diagram as PNG
52 os.system(f"plantuml {uml_file_path}")
53
54 # Display the generated diagram
55 from IPython.display import Image
56 Image(filename="/content/sequence_diagram.png")
57
58 Reading package lists... Done
59 Building dependency tree... Done
60 Reading state information... Done
61 plantuml is already the newest version (1:1.2020.2+ds-1).
62 0 upgraded, 0 newly installed, 0 to remove and 29 not upgraded.

```



```

1 # Install PlantUML (if not already installed)
2 !apt-get install -y plantuml
3
4 # Import necessary modules
5 import os
6 from IPython.display import Image, display

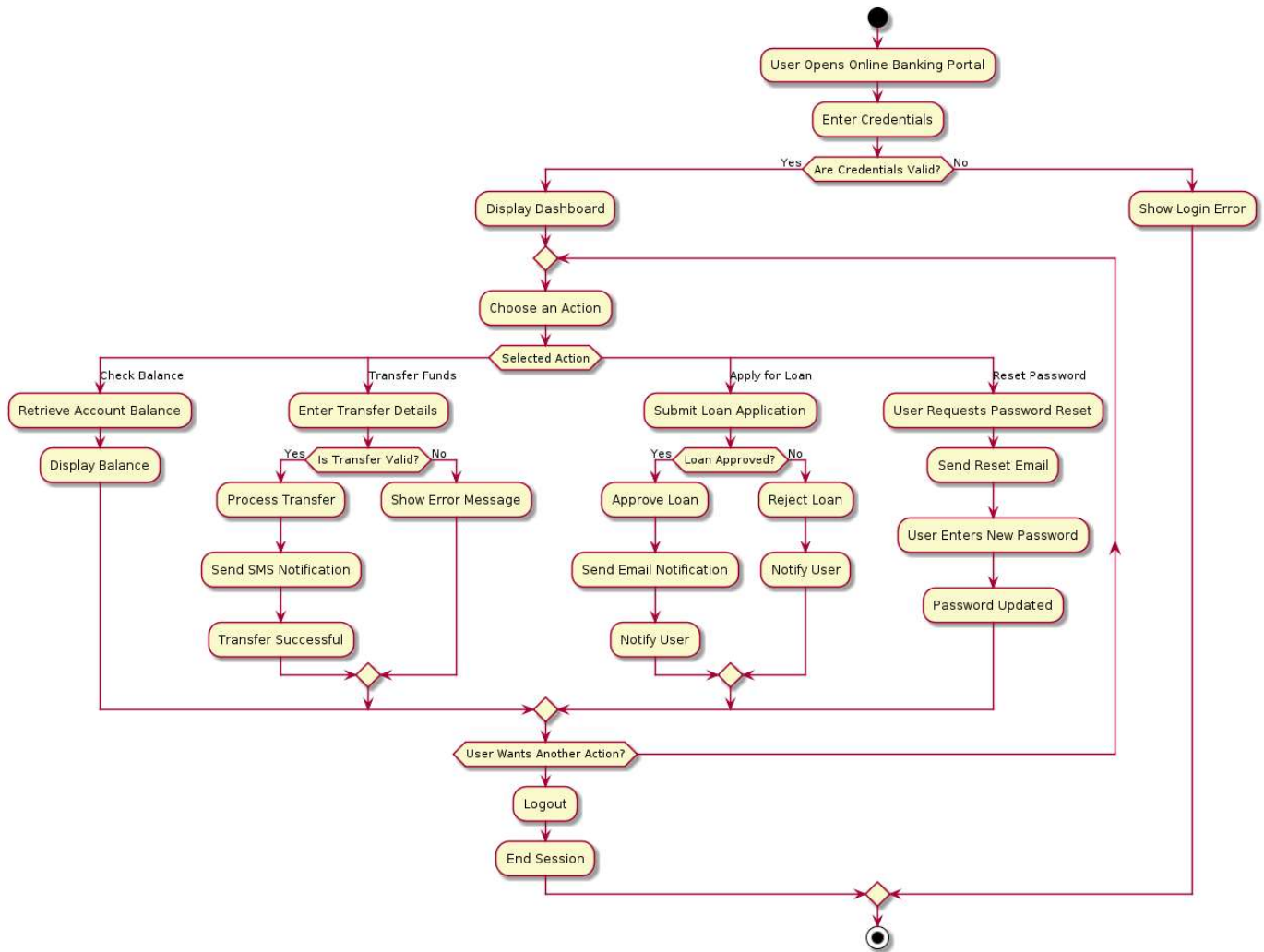
```

```

7
8 # PlantUML code for the activity diagram
9 uml_code = ""
10 @startuml
11 start
12 :User Opens Online Banking Portal;
13 :Enter Credentials;
14 if (Are Credentials Valid?) then (Yes)
15     :Display Dashboard;
16     repeat
17         :Choose an Action;
18         switch (Selected Action)
19             case (Check Balance)
20                 :Retrieve Account Balance;
21                 :Display Balance;
22             case (Transfer Funds)
23                 :Enter Transfer Details;
24                 if (Is Transfer Valid?) then (Yes)
25                     :Process Transfer;
26                     :Send SMS Notification;
27                     :Transfer Successful;
28                 else (No)
29                     :Show Error Message;
30                 endif
31             case (Apply for Loan)
32                 :Submit Loan Application;
33                 if (Loan Approved?) then (Yes)
34                     :Approve Loan;
35                     :Send Email Notification;
36                     :Notify User;
37                 else (No)
38                     :Reject Loan;
39                     :Notify User;
40                 endif
41             case (Reset Password)
42                 :User Requests Password Reset;
43                 :Send Reset Email;
44                 :User Enters New Password;
45                 :Password Updated;
46             endswitch
47         repeat while (User Wants Another Action?)
48
49         :Logout;
50         :End Session;
51     else (No)
52         :Show Login Error;
53     endif
54 stop
55 @enduml
56 ""
57
58 # Define file paths
59 uml_file_path = "/content/activity_diagram.puml"
60 output_image_path = "/content/activity_diagram.png"
61
62 # Create a temporary file for the UML code
63 with open(uml_file_path, "w") as file:
64     file.write(uml_code)
65
66 # Generate the PlantUML diagram as PNG
67 os.system(f"plantuml -tpng {uml_file_path}")
68
69 # Check if the image was generated successfully
70 if os.path.exists(output_image_path):
71     print("✅ Activity Diagram Generated Successfully!")
72     display(Image(filename=output_image_path))
73 else:
74     print("❌ Error: Failed to generate the activity diagram.")
75

```

Reading package lists... Done  
 Building dependency tree... Done  
 Reading state information... Done  
 plantuml is already the newest version (1:1.2020.2+ds-1).  
 0 upgraded, 0 newly installed, 0 to remove and 29 not upgraded.  
 Activity Diagram Generated Successfully!




```

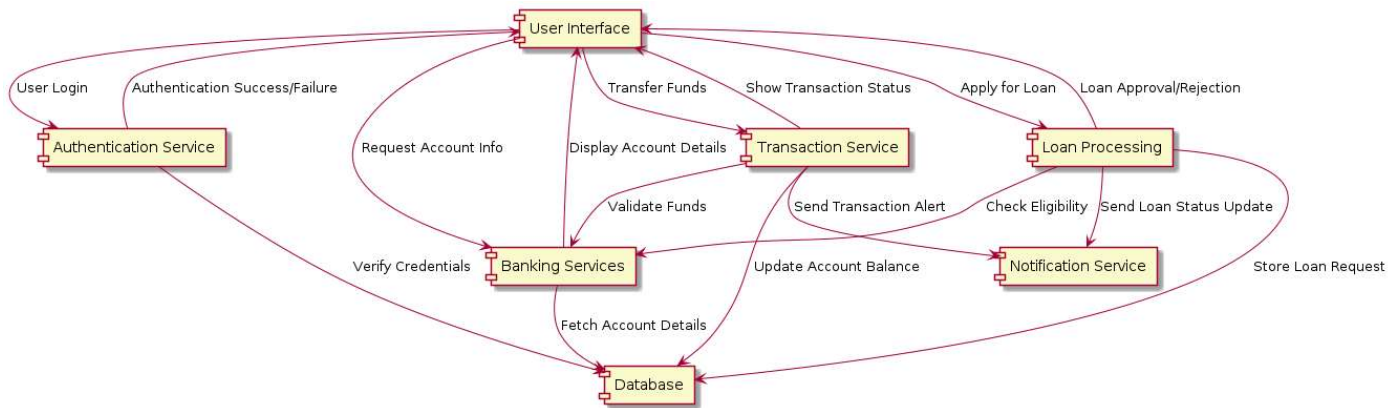
1 # Install PlantUML (if not already installed)
2 !apt-get install -y plantuml
3
4 # Import necessary modules
5 import os
6 from IPython.display import Image, display
7
8 # PlantUML code for the component diagram
9 uml_code = ""
10 @startuml
11 component "User Interface" as UI
12 component "Authentication Service" as Auth
13 component "Banking Services" as Banking
14 component "Transaction Service" as Transaction
15 component "Loan Processing" as Loan
16 component "Notification Service" as Notify
17 component "Database" as DB
18
19 UI --> Auth : User Login
20 Auth --> DB : Verify Credentials
  
```

```

21 Auth --> UI : Authentication Success/Failure
22 UI --> Banking : Request Account Info
23 Banking --> DB : Fetch Account Details
24 Banking --> UI : Display Account Details
25 UI --> Transaction : Transfer Funds
26 Transaction --> Banking : Validate Funds
27 Transaction --> DB : Update Account Balance
28 Transaction --> Notify : Send Transaction Alert
29 Transaction --> UI : Show Transaction Status
30 UI --> Loan : Apply for Loan
31 Loan --> Banking : Check Eligibility
32 Loan --> DB : Store Loan Request
33 Loan --> Notify : Send Loan Status Update
34 Loan --> UI : Loan Approval/Rejection
35 @enduml
36 """
37
38 # Define file paths
39 uml_file_path = "/content/component_diagram.puml"
40 output_image_path = "/content/component_diagram.png"
41
42 # Create a temporary file for the UML code
43 with open(uml_file_path, "w") as file:
44     file.write(uml_code)
45
46 # Generate the PlantUML diagram as PNG
47 os.system(f"plantuml -tpng {uml_file_path}")
48
49 # Check if the image was generated successfully
50 if os.path.exists(output_image_path):
51     print("✅ Component Diagram Generated Successfully!")
52     display(Image(filename=output_image_path))
53 else:
54     print("❌ Error: Failed to generate the component diagram.")
55

```

 Reading package lists... Done  
 Building dependency tree... Done  
 Reading state information... Done  
 plantuml is already the newest version (1:1.2020.2+ds-1).  
 0 upgraded, 0 newly installed, 0 to remove and 29 not upgraded.  
 ✅ Component Diagram Generated Successfully!



```

1 # Install PlantUML (if not already installed)
2 !apt-get install -y plantuml
3
4 # Import necessary modules
5 import os
6 from IPython.display import Image, display
7
8 # PlantUML code for the component diagram with nodes
9 uml_code = """
10 @startuml
11 node "User Device" {
12     component "Web Browser" as Browser
13 }
14
15 node "Bank Server" {

```



```

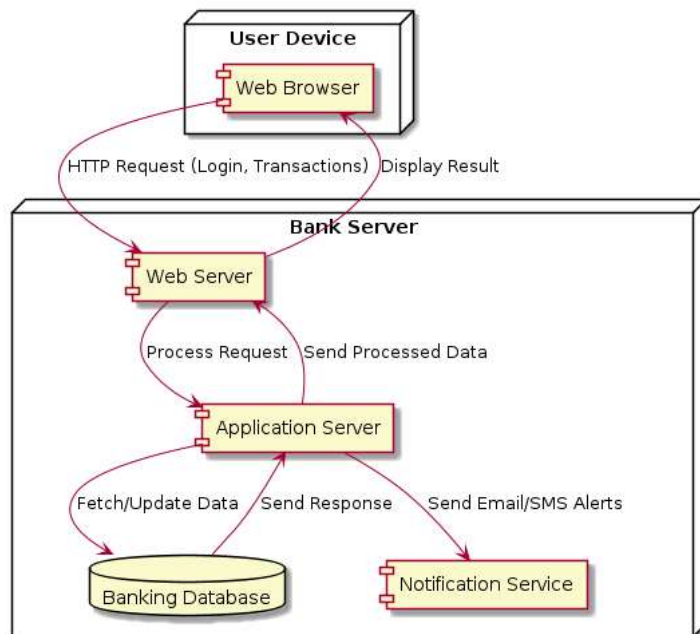
16 component "Web Server" as WebServer
17 component "Application Server" as AppServer
18 database "Banking Database" as DB
19 component "Notification Service" as Notify
20 }
21
22 Browser --> WebServer : HTTP Request (Login, Transactions)
23 WebServer --> AppServer : Process Request
24 AppServer --> DB : Fetch/Update Data
25 DB --> AppServer : Send Response
26 AppServer --> Notify : Send Email/SMS Alerts
27 AppServer --> WebServer : Send Processed Data
28 WebServer --> Browser : Display Result
29 @enduml
30 """
31
32 # Define file paths
33 uml_file_path = "/content/component_with_nodes_diagram.puml"
34 output_image_path = "/content/component_with_nodes_diagram.png"
35
36 # Create a temporary file for the UML code
37 with open(uml_file_path, "w") as file:
38     file.write(uml_code)
39
40 # Generate the PlantUML diagram as PNG
41 os.system(f"plantuml -tpng {uml_file_path}")
42
43 # Check if the image was generated successfully
44 if os.path.exists(output_image_path):
45     print("✅ Component Diagram with Nodes Generated Successfully!")
46     display(Image(filename=output_image_path))
47 else:
48     print("❌ Error: Failed to generate the component diagram.")
49

```

```

🔄 Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
plantuml is already the newest version (1:1.2020.2+ds-1).
0 upgraded, 0 newly installed, 0 to remove and 29 not upgraded.
✅ Component Diagram with Nodes Generated Successfully!

```



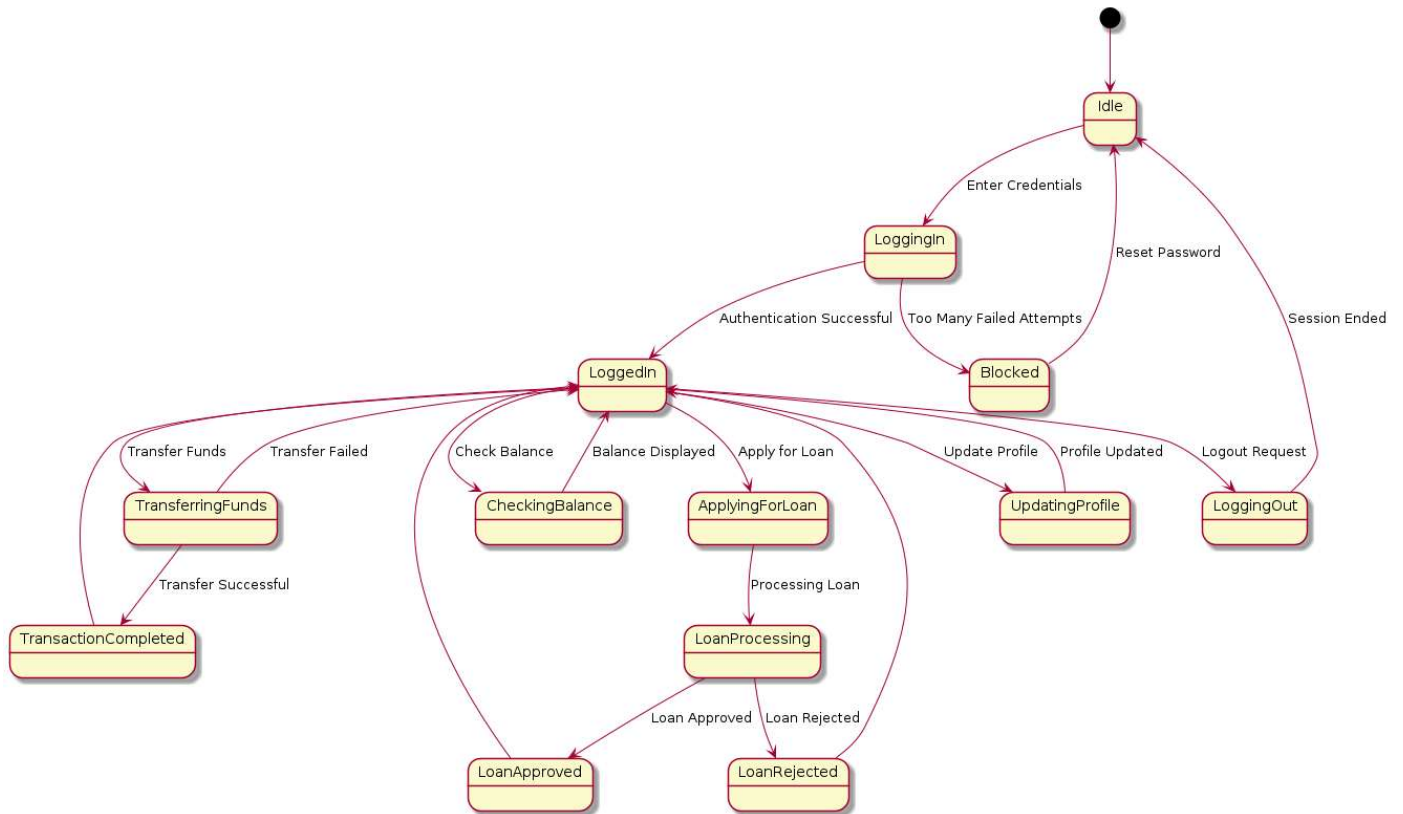
```

1 # Install PlantUML (if not already installed)
2 !apt-get install -y plantuml
3
4 # Import necessary modules
5 import os
6 from IPython.display import Image, display
7
8 # PlantUML code for the state diagram
9 uml_code = """

```

```
10 @startuml
11 [*] --> Idle
12 Idle --> LoggingIn : Enter Credentials
13 LoggingIn --> LoggedIn : Authentication Successful
14 LoggingIn --> Blocked : Too Many Failed Attempts
15 Blocked --> Idle : Reset Password
16 LoggedIn --> CheckingBalance : Check Balance
17 CheckingBalance --> LoggedIn : Balance Displayed
18 LoggedIn --> TransferringFunds : Transfer Funds
19 TransferringFunds --> TransactionCompleted : Transfer Successful
20 TransferringFunds --> LoggedIn : Transfer Failed
21 LoggedIn --> ApplyingForLoan : Apply for Loan
22 ApplyingForLoan --> LoanProcessing : Processing Loan
23 LoanProcessing --> LoanApproved : Loan Approved
24 LoanProcessing --> LoanRejected : Loan Rejected
25 LoanApproved --> LoggedIn
26 LoanRejected --> LoggedIn
27 TransactionCompleted --> LoggedIn
28 LoggedIn --> UpdatingProfile : Update Profile
29 UpdatingProfile --> LoggedIn : Profile Updated
30 LoggedIn --> LoggingOut : Logout Request
31 LoggingOut --> Idle : Session Ended
32 @enduml
33 ""
34
35 # Define file paths
36 uml_file_path = "/content/state_diagram.puml"
37 output_image_path = "/content/state_diagram.png"
38
39 # Create a temporary file for the UML code
40 with open(uml_file_path, "w") as file:
41     file.write(uml_code)
42
43 # Generate the PlantUML diagram as PNG
44 os.system(f"plantuml -tpng {uml_file_path}")
45
46 # Check if the image was generated successfully
47 if os.path.exists(output_image_path):
48     print("✅ State Diagram Generated Successfully!")
49     display(Image(filename=output_image_path))
50 else:
51     print("❌ Error: Failed to generate the state diagram.")
52
```

Reading package lists... Done  
 Building dependency tree... Done  
 Reading state information... Done  
 plantuml is already the newest version (1:1.2020.2+ds-1).  
 0 upgraded, 0 newly installed, 0 to remove and 29 not upgraded.  
 State Diagram Generated Successfully!



```

1 # Install PlantUML (if not already installed)
2 !apt-get install -y plantuml
3
4 # Import necessary modules
5 import os
6 from IPython.display import Image, display
7
8 # PlantUML code for the collaboration diagram
9 uml_code = ""
10 @startuml
11 title Collaboration Diagram for Secure Online Banking System
12
13 object Customer
14 object BankingSystem
15 object Account
16 object Transaction
17 object Loan
18 object BankEmployee
19 object NotificationService
20
21 ' User login process
22 Customer --> BankingSystem : (1) Login Request
23 BankingSystem --> Account : (2) Validate Credentials
24 Account --> BankingSystem : (3) Authentication Success
25 BankingSystem --> Customer : (4) Login Successful
26 BankingSystem --> Customer : (5) Login Failed
27 Customer --> BankingSystem : (6) Reset Password Request
28
29 ' Checking balance
30 Customer --> BankingSystem : (7) Request Account Balance
31 BankingSystem --> Account : (8) Retrieve Balance
  
```

```
32 Account --> BankingSystem : (9) Send Balance Info
33 BankingSystem --> Customer : (10) Display Balance
34
35 ' Fund transfer
36 Customer --> BankingSystem : (11) Initiate Fund Transfer
37 BankingSystem --> Transaction : (12) Process Transfer
38 Transaction --> Account : (13) Debit Source Account
39 Transaction --> Account : (14) Credit Destination Account
40 Transaction --> BankingSystem : (15) Transfer Confirmation
41 BankingSystem --> NotificationService : (16) Send Transaction Alert
42 NotificationService --> Customer : (17) SMS Notification Sent
43
44 ' Loan Application
45 Customer --> BankingSystem : (18) Apply for Loan
46 BankingSystem --> Loan : (19) Process Loan Request
47 Loan --> BankEmployee : (20) Request Approval
```