| **UC Name** | **UC\_Admin\_13 – Monitor All Transactions** |
| --- | --- |

|  |  |
| --- | --- |
| **Summary** | Administrator monitors all transactions in the system for auditing, security, or analysis purposes. |

|  |  |
| --- | --- |
| **Dependency** | Requires successful login and admin privileges. |

|  |  |
| --- | --- |
| **Actors** | **Primary Actor**: Administrator **Secondary Actor**: System (to retrieve and display logs) |

|  |  |
| --- | --- |
| **Preconditions** | • The Administrator must be authenticated. • Logging system must be active. • Admin panel must be accessible. |

|  |  |
| --- | --- |
| **Description of the Main Sequence** | • Step 1: Administrator logs into the system. • Step 2: Navigates to the “Transaction Monitor” section in the admin panel. • Step 3: Applies filters such as date, transaction type, amount, or user ID. • Step 4: The system queries and retrieves relevant transactions. • Step 5: Administrator views transaction results. • Step 6: Admin selects a transaction for full details or flags one for review. |

|  |  |
| --- | --- |
| **Description of the Alternative Sequence** | • Step 1: Admin skips filters and views all transactions (default view). • Step 2: If result is too large, the system prompts to refine filters. • Step 3: Admin may switch to real-time monitoring mode. |

|  |  |
| --- | --- |
| **Non functional requirements** | • Transactions must load in under 2 seconds. • Data must be encrypted. • The interface should support sorting, filtering, and export to PDF/CSV. |

|  |  |
| --- | --- |
| **Postconditions** | • Admin successfully views, exports, or flags transactions. • All admin monitoring activity is logged for auditing. |

@startuml

actor Administrator

usecase "Login to System" as UC1

usecase "Access Transaction Monitoring Module" as UC2

usecase "Filter Transactions by Criteria" as UC3

usecase "View Transaction List" as UC4

usecase "Open Transaction Details" as UC5

Administrator --> UC1

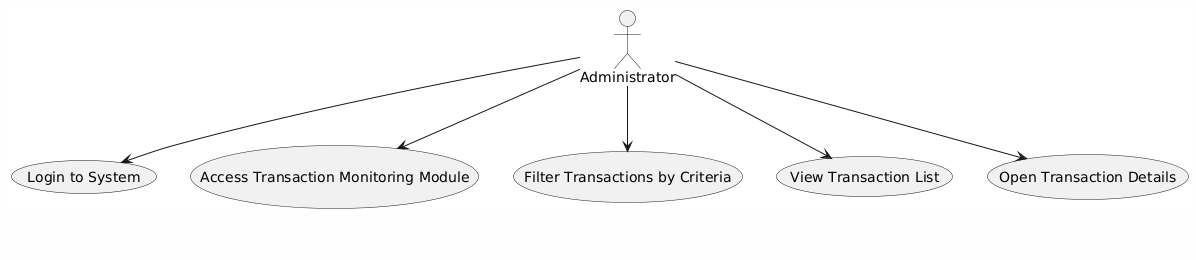
Administrator --> UC2

Administrator --> UC3

Administrator --> UC4

Administrator --> UC5

@enduml



| **UC Name** | **UC\_Admin\_14 – Freeze or Close Accounts** |
| --- | --- |

|  |  |
| --- | --- |
| **Summary** | Administrator freezes or closes customer accounts due to suspicious activity or based on a customer request. |

|  |  |
| --- | --- |
| **Dependency** | Requires successful login and access to account management tools. |

|  |  |
| --- | --- |
| **Actors** | **Primary Actor**: Administrator **Secondary Actor**: Bank System |

|  |  |
| --- | --- |
| **Preconditions** | • The administrator must be authenticated. • Customer account must exist and be active. • System must allow modification of account status. |

|  |  |
| --- | --- |
| **Description of the Main Sequence** | ● Step 1: Administrator logs into the Bank System. ● Step 2: Searches for the customer account to manage. ● Step 3: Chooses the option to either freeze or close the account. ● Step 4: System validates the action based on account state and business rules. ● Step 5: Administrator confirms the action. ● Step 6: Bank System updates the account status accordingly and logs the event. |

|  |  |
| --- | --- |
| **Description of the Alternative Sequence** | ● Step 1: If account is already closed, the system displays an error. ● Step 2: If account is under investigation, freeze/close option is temporarily locked. ● Step 3: Admin can cancel the action before confirming. |

|  |  |
| --- | --- |
| **Non functional requirements** | • Status change should reflect instantly across the system. • All actions must be logged securely with timestamps. • Interface should prevent accidental clicks via confirmation prompt. |

|  |  |
| --- | --- |
| **Postconditions** | • The account is marked as "Frozen" or "Closed". • Notifications are sent to the customer. • Action is logged in the admin activity logs. |

@startuml

actor Administrator

usecase "Login to System" as UC1

usecase "Access Account Management" as UC2

usecase "Search for Account" as UC3

usecase "Freeze Account" as UC4

usecase "Close Account" as UC5

usecase "Confirm Action and Provide Reason" as UC6

Administrator --> UC1

Administrator --> UC2

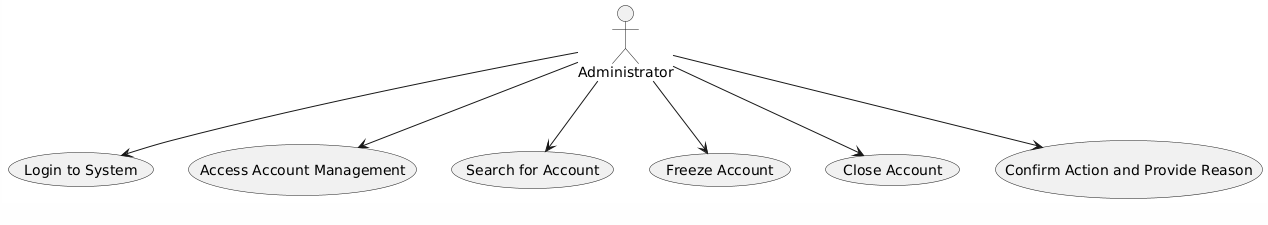
Administrator --> UC3

Administrator --> UC4

Administrator --> UC5

Administrator --> UC6

@enduml



| **UC Name** | **UC\_Admin\_15 – Generate Transaction Reports** |
| --- | --- |

|  |  |
| --- | --- |
| **Summary** | Administrator generates reports based on transaction data for analysis, auditing, or exporting purposes. |

|  |  |
| --- | --- |
| **Dependency** | Depends on transaction data being logged and available in the system. |

|  |  |
| --- | --- |
| **Actors** | **Primary Actor**: Administrator **Secondary Actor**: Bank System |

|  |  |
| --- | --- |
| **Preconditions** | • The Administrator must be logged in and authorized to generate reports. • Transaction data must be available and up-to-date. • System must have reporting functionality enabled. |

|  |  |
| --- | --- |
| **Description of the Main Sequence** | ● Step 1: Administrator logs into the system. ● Step 2: Navigates to the "Reports" section. ● Step 3: Selects the type of transaction report (daily, monthly, by user, etc). ● Step 4: Applies relevant filters such as date range or amount. ● Step 5: System fetches and compiles transaction data. ● Step 6: Administrator previews and downloads/export the report in desired format. |

|  |  |
| --- | --- |
| **Description of the Alternative Sequence** | ● Step 1: If no filters are applied, system defaults to showing the last 30 days. ● Step 2: If report size is large, system compresses output. ● Step 3: Admin cancels report generation mid-process. |

|  |  |
| --- | --- |
| **Non functional requirements** | • Reports must generate within 5 seconds. • Export formats must include PDF and CSV. • All generated reports should be logged for future access. |

|  |  |
| --- | --- |
| **Postconditions** | • Report is successfully generated and saved or downloaded. • System logs the reporting activity. |

@startuml

actor Administrator

usecase "Login to System" as UC1

usecase "Navigate to Reports Section" as UC2

usecase "Select Transaction Reports" as UC3

usecase "Set Report Parameters" as UC4

usecase "Generate Report" as UC5

usecase "Download Report (PDF/Excel)" as UC6

Administrator --> UC1

Administrator --> UC2

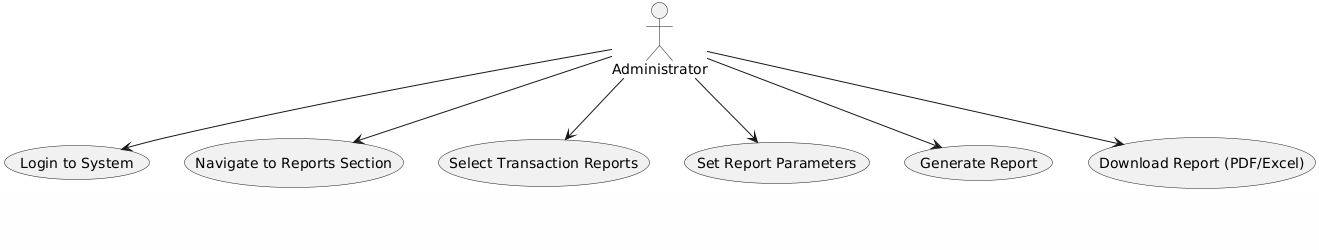
Administrator --> UC3

Administrator --> UC4

Administrator --> UC5

Administrator --> UC6

@enduml



| **UC Name** | **UC\_System\_16 – Validate Account Details During Creation** |
| --- | --- |

|  |  |
| --- | --- |
| **Summary** | The system checks and verifies all the information provided by the customer when creating a new account. |

|  |  |
| --- | --- |
| **Dependency** | Depends on the customer filling out the registration form and submitting valid input. |

|  |  |
| --- | --- |
| **Actors** | **Primary Actor**: Customer **Secondary Actor**: Bank System |

|  |  |
| --- | --- |
| **Preconditions** | • Customer has started the registration process. • All required fields in the account creation form are visible. • Backend validation services are running. |

|  |  |
| --- | --- |
| **Description of the Main Sequence** | ● Step 1: Customer accesses the account creation page. ● Step 2: Enters required personal details (name, DOB, email, phone, etc). ● Step 3: Uploads necessary KYC documents. ● Step 4: Submits the form. ● Step 5: Bank System validates format (e.g., valid email, phone). ● Step 6: Bank System checks for existing duplicate records. ● Step 7: System verifies uploaded documents for readability and correctness. ● Step 8: Validation is either passed or failed and feedback is shown. |

|  |  |
| --- | --- |
| **Description of the Alternative Sequence** | ● Step 1: If fields are missing, system highlights them in red. ● Step 2: If duplicate email or ID exists, system prompts the user to choose a different one. ● Step 3: If document is unreadable, system requests a new upload. |

|  |  |
| --- | --- |
| **Non functional requirements** | • Validation should complete within 3 seconds. • Form must be mobile-friendly. • All invalid fields must have helpful error messages. |

|  |  |
| --- | --- |
| **Postconditions** | • If successful, the customer moves to the next step (account confirmation). • All validation logs are stored for auditing. |

@startuml

actor Customer

usecase "Open Account Creation Form" as UC1

usecase "Enter Account Details" as UC2

usecase "Submit Account Creation Form" as UC3

usecase "System Validates Input" as UC4

usecase "Account is Created Successfully" as UC5

Customer --> UC1

Customer --> UC2

Customer --> UC3

Customer --> UC4

Customer --> UC5

@enduml

