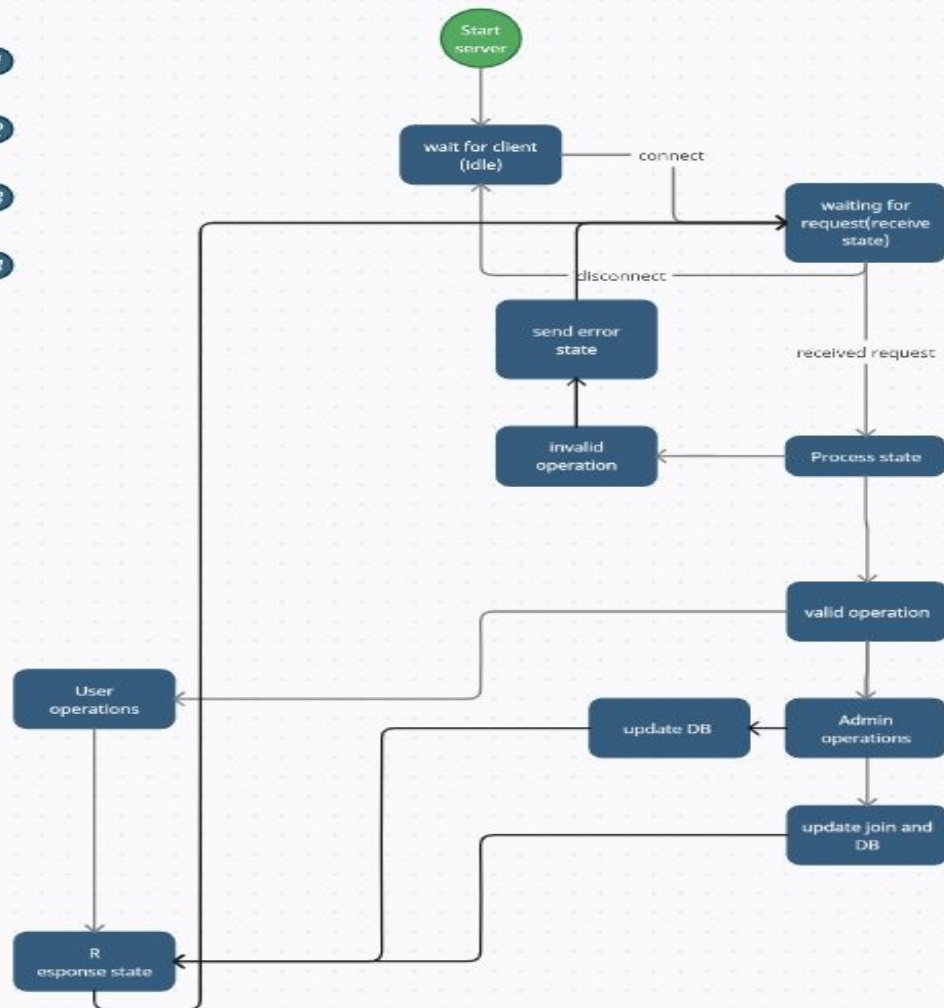


# Bank management system

Server state diagram

Server state diagram

1  
2  
3  
4



### Server state diagram

```

graph LR
    A[Idle] --> B[Client Connected]
    B --> C[Receive Data]
    C --> D[Process Data]
    D --> E[Valid Operation]
    D --> F[Invalid Operation]
    E --> G[Update Database] [User]
    E --> H[Update Both Databases] [Admin]
    G --> I[Send Response]
    H --> I[Send Response]
    I --> B
    F --> J[Send Error Message]
    J --> B
    B --> K[Client Disconnected]
    K --> A
    A --> L[Thread Terminated]
    subgraph User Operations
        E --> M[Check Balance/Transaction]
        M --> N[Send Response]
        N --> I
        E --> O[Perform Deposit/Withdrawal]
        O --> P[Update Balance]
        P --> G
    end
    end
    subgraph Admin Operations
        E --> Q[View All Accounts]
        Q --> R[Send Database]
        R --> I
        E --> S[Delete User]
        S --> T[Delete User Data] [Login & Bank DB]
        T --> I
        E --> U[Update User]
        U --> V[Update User Data] [Login & Bank DB]
        V --> I
    end
    end
  
```

The diagram lists the states and transitions for the server. States are represented by letters A through V. Transitions are labeled with actions and sometimes the user type (User or Admin). The diagram is divided into sections for User Operations and Admin Operations. The 'end' keyword is used to mark the end of the state diagram.

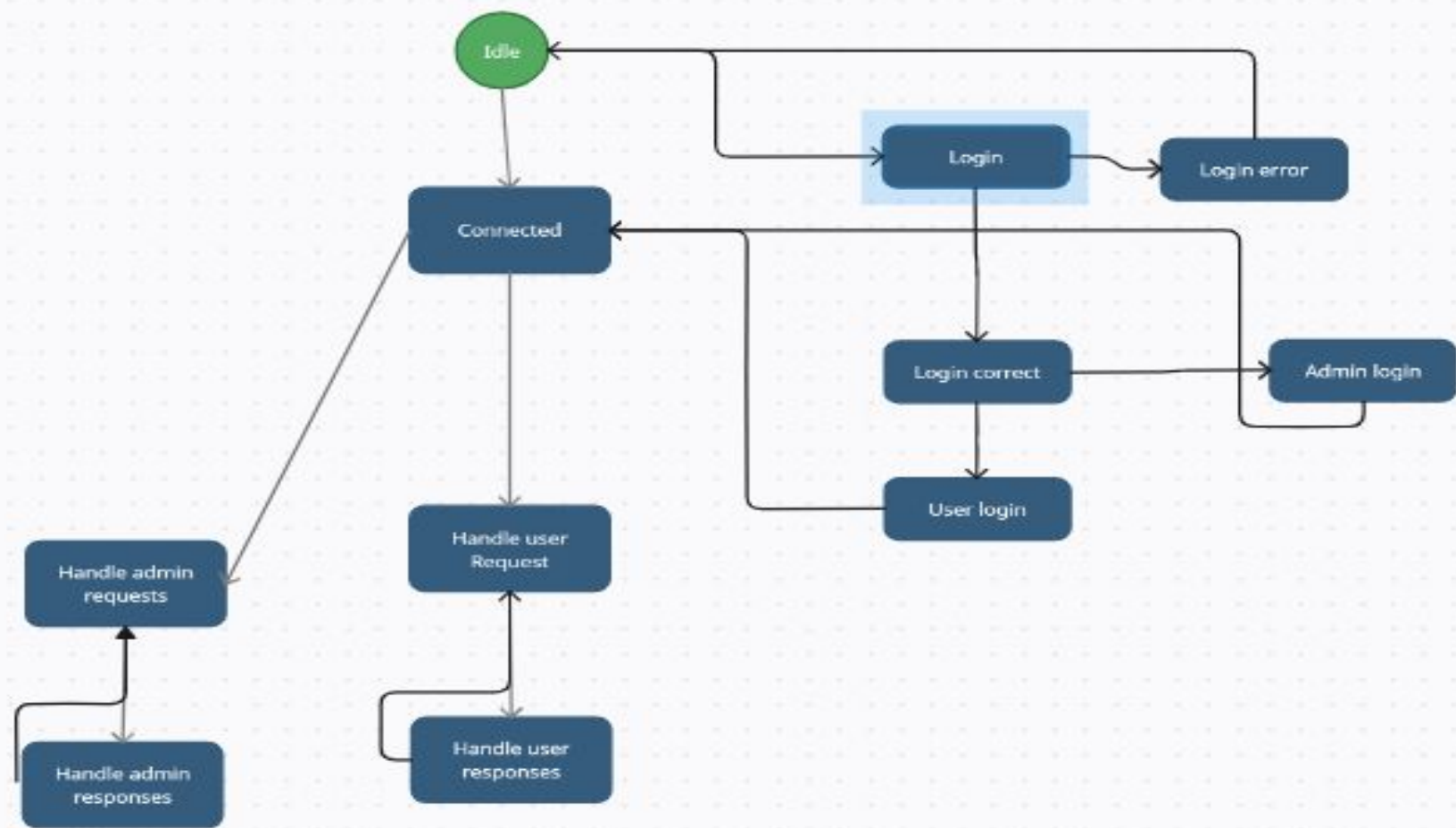
tag1

tag2

...

Client state diagram

Server state diagram



Server class diagram

Server state diagram



signals

interacts

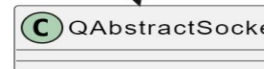
uses

handles

listens to events

**MainWindow**

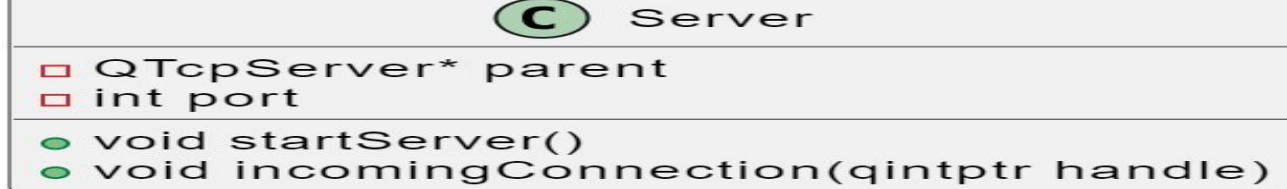
- MainWindow(QWidget \*parent)
- ~MainWindow()
- void onConnectionDevice()
- void onDisconnectedDevice()
- void onErrorOccurredDevice(QAbstractSocket::SocketError socketError)
- void onStateChangedDevice(QAbstractSocket::SocketState socketState)
- void onReadyReadDevice(QString data)
- void on\_pbConnect\_clicked()
- void on\_pbDisconnect\_clicked()
- void on\_pbSend\_clicked()
- void on\_pushButton\_clicked()
- void on\_pushButton\_11\_clicked()
- void on\_pushButton\_10\_clicked()
- void on\_pushButton\_9\_clicked()
- void on\_pushButton\_13\_clicked()
- void on\_pushButton\_12\_clicked()
- void on\_pushButton\_2\_clicked()
- void on\_pushButton\_8\_clicked()
- void on\_pushButton\_5\_clicked()
- void on\_pushButton\_4\_clicked()
- void on\_pushButton\_7\_clicked()
- void on\_pushButton\_3\_clicked()
- void on\_pushButton\_6\_clicked()



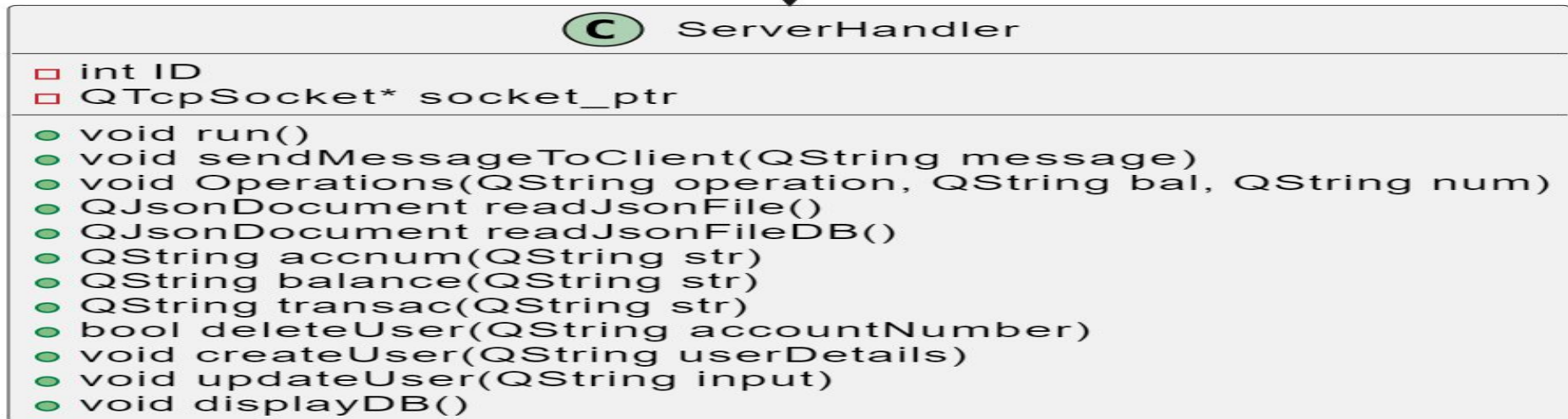
Client class diagram

Server state diagram





Manages  
1..\*



Uses

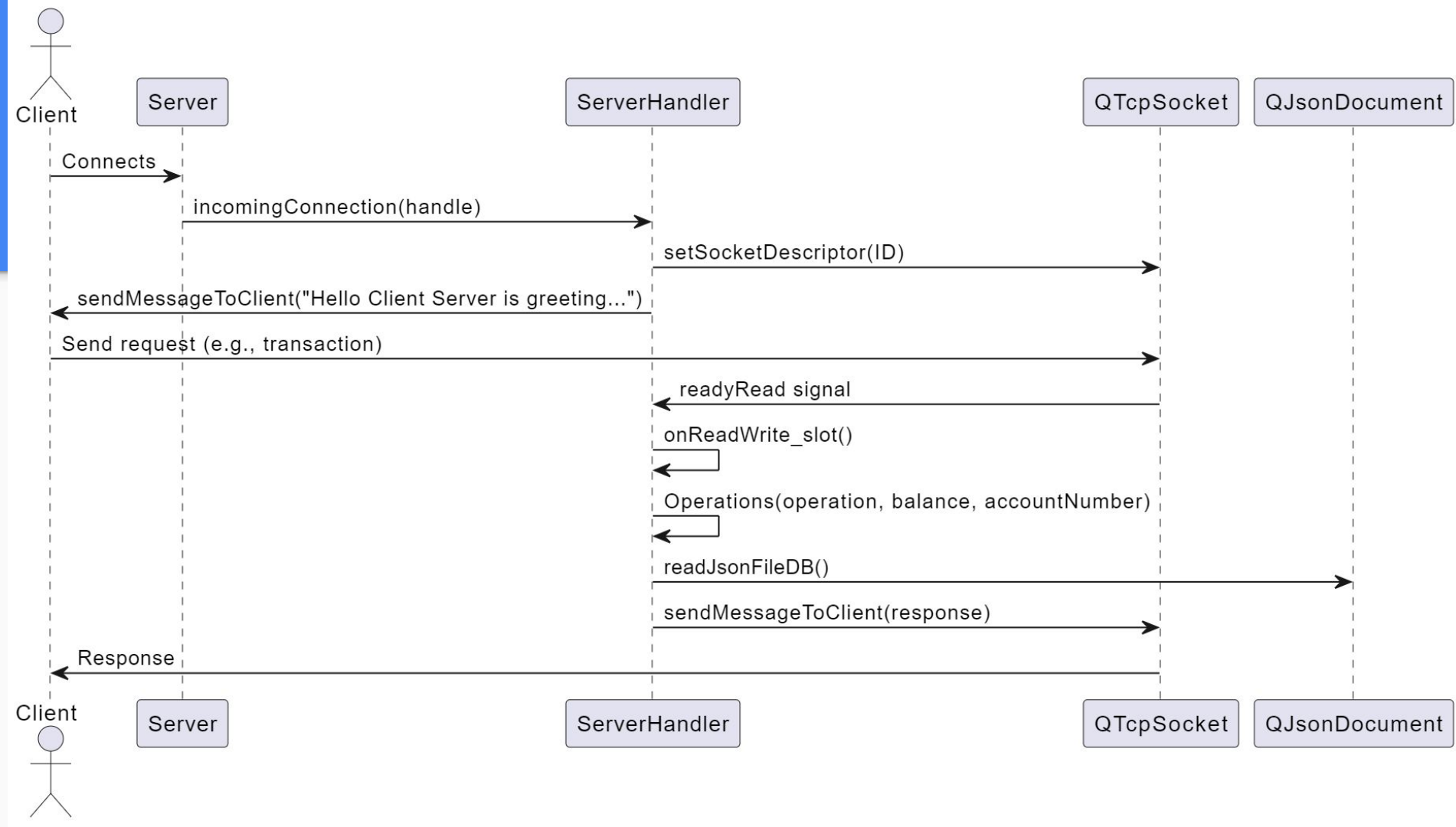
Reads/Writes

Accesses



Server sequence diagram

Server state diagram



Client sequence diagram

Server state diagram

