UMLs: Sequence Diagrams

Tags

- Similiar to Class Digrams Sequence diagrams show how classes within a system or objects within code interact with each other

- In particular sequence diagrams show the order of the events

Example: ATM

- A ATM transaction involves the folloiwng steps:

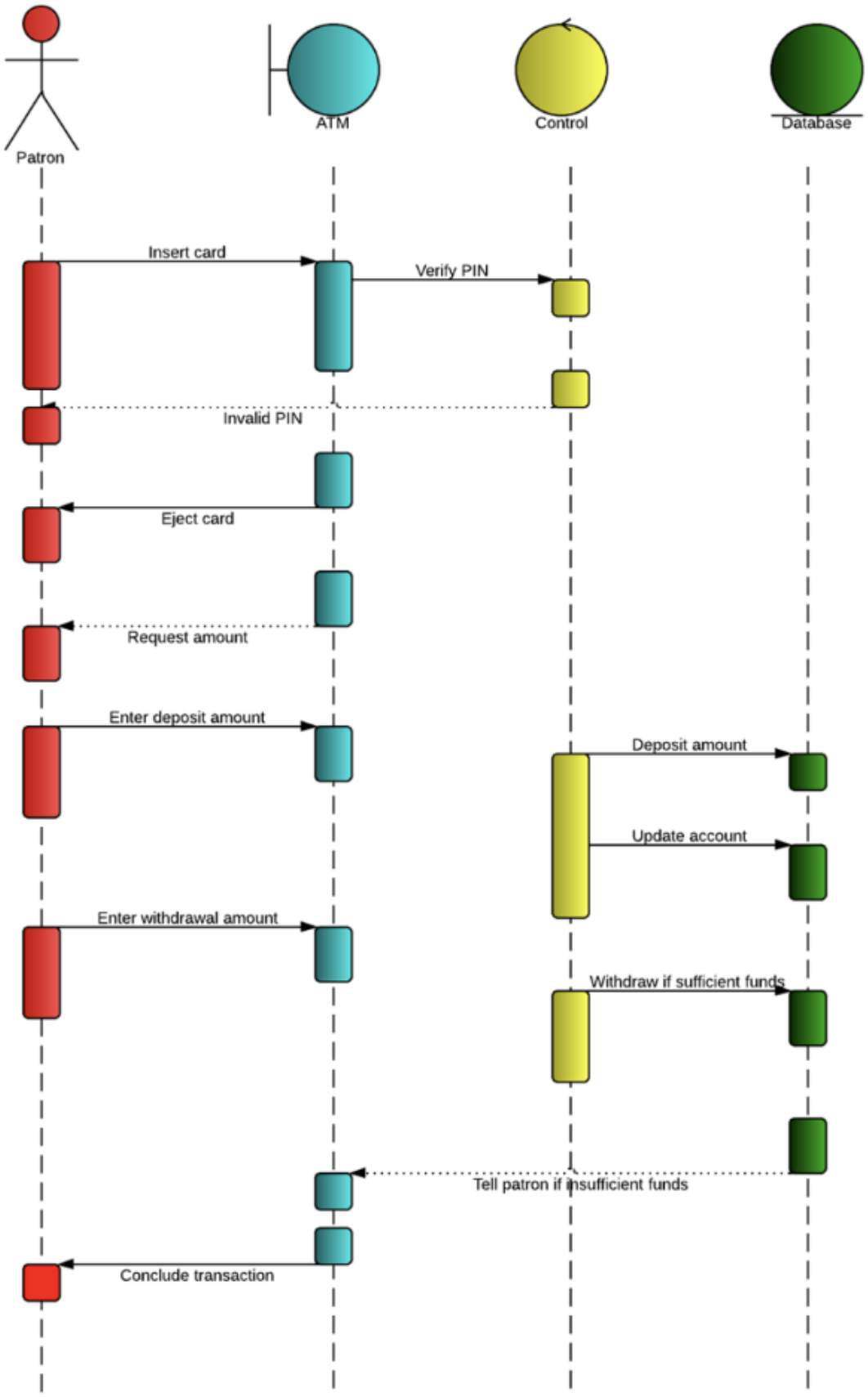
Person → ATM → Bank Server → Bank Account

- A person will use the ATM to access the Banks Server, which will then allow the person access to their own bank account
- In this example the person is called a **actor**
- The ATM / Bank Server / Bank Account are the **objects**

- The the sequence of events go from left to right.
- The dotted lines are called the **lifelines** these show the existance of an object or an actor over time, i.e. moving down the lifeline shows more time is passing

Sequence of events for an ATM transaction as represented by the Sequence Diagram above:

- The first **message** is the person interacting with the ATM by inserting thier card
- The **second message** is the verification of the bank card which happens between the ATM and the bank server
- The **retrun message** is shown by the dotted arrows here the return message shows that the bank card was verfieid ok
- However this is an **alternate case**, where if the card is valid or not, shown by the condition statemetns
- This then triggers the ATM to ask for a pin. But notice this is not a dotted line, because this is not a reply message directly realted to a another request.
- Then the ATM asks the bank server to verify pin
- An alternate case is again created
- If the pin and the card are valid the user is asked to enter a amount to withdraw,. again creating an alternate case, to see if the funds are sifficiant
- The boxes around the dotted lines are the **Activation boxes** these show the activation period of the objects


Patron
ATM
Control
Database
Insert card
Verify PIN
Invalid PIN
Eject card
Request amount
Enter deposit amount
Deposit amount
Update account
Enter withdrawal amount
Withdraw if sufficient funds
Tell patron if insufficient funds
Conclude transaction

Common Message Symbols:

Synchronous message

- Represented by a solid line with a solid arrowhead. This symbol is used when the sender must await for a response before continuing

Asynchronous message

- These messages do not require a response before continuing

Asynchronous retrun message

- Represented by a dashed line, these messages are response or return messages to the synchronous messages.

Use cases for sequence diagrams:

- Usage scenario with a clear timeline of events
- Method logic: Explore the logic of a system