

# Unified Modeling Language

Currency exchange website

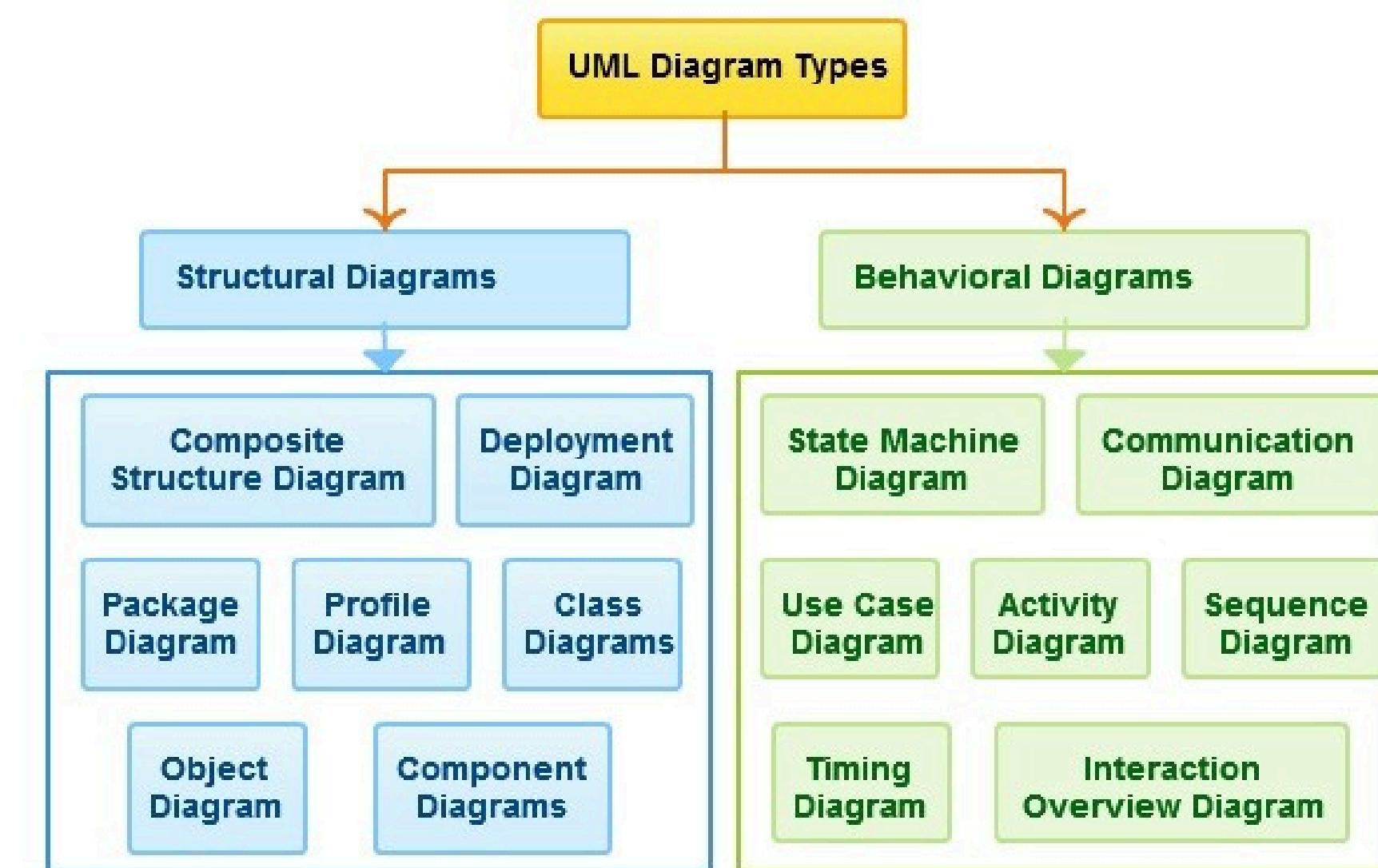
UML Diagram

Katarzyna Brzeski



KB

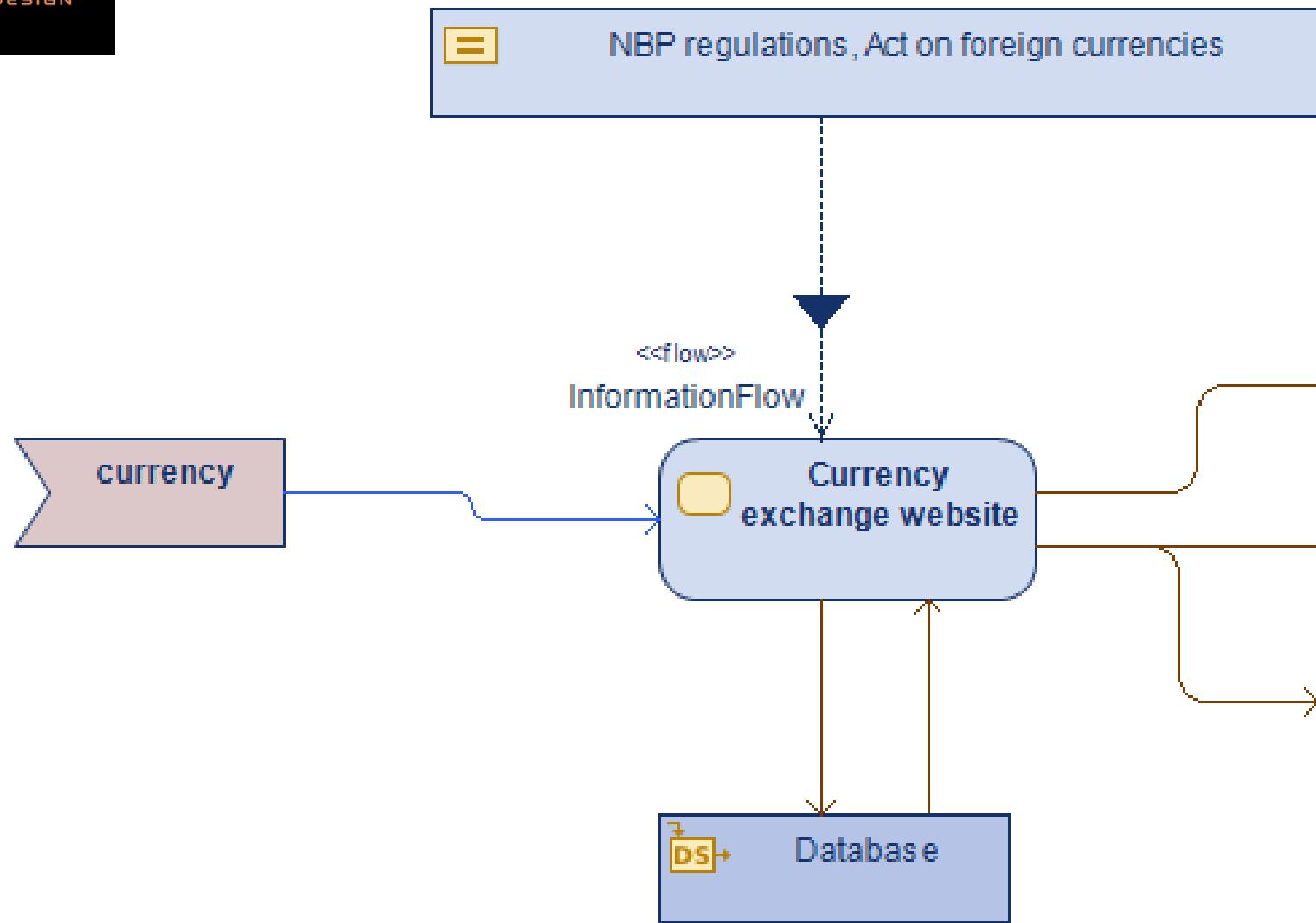
SCIENCE & DESIGN



# Types of diagrams



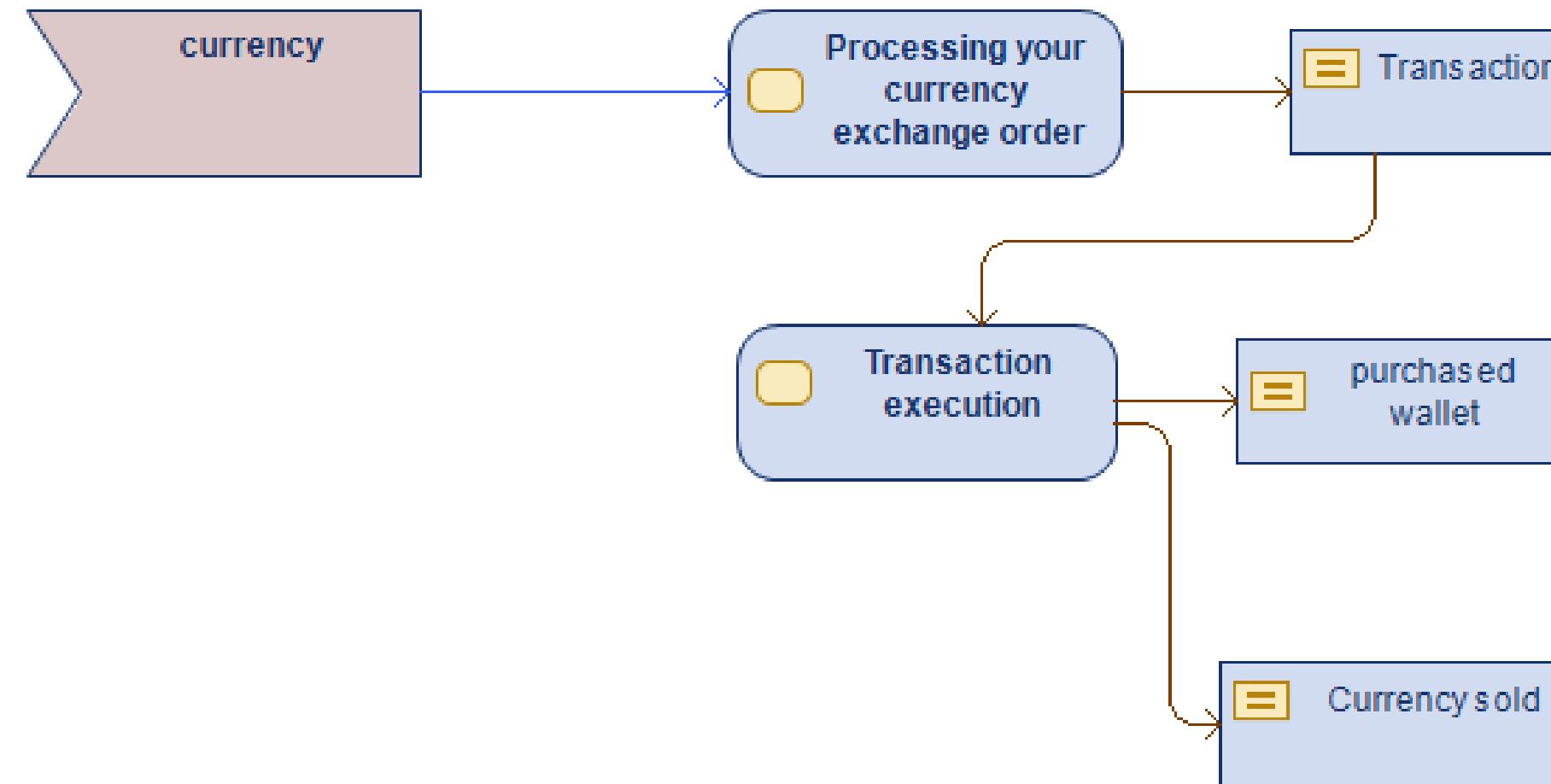
KB  
SCIENCE & DESIGN



A context diagram It shows the main information that is processed by the system and the main products and services of the system - it models the main process of the system using a UML activity diagram

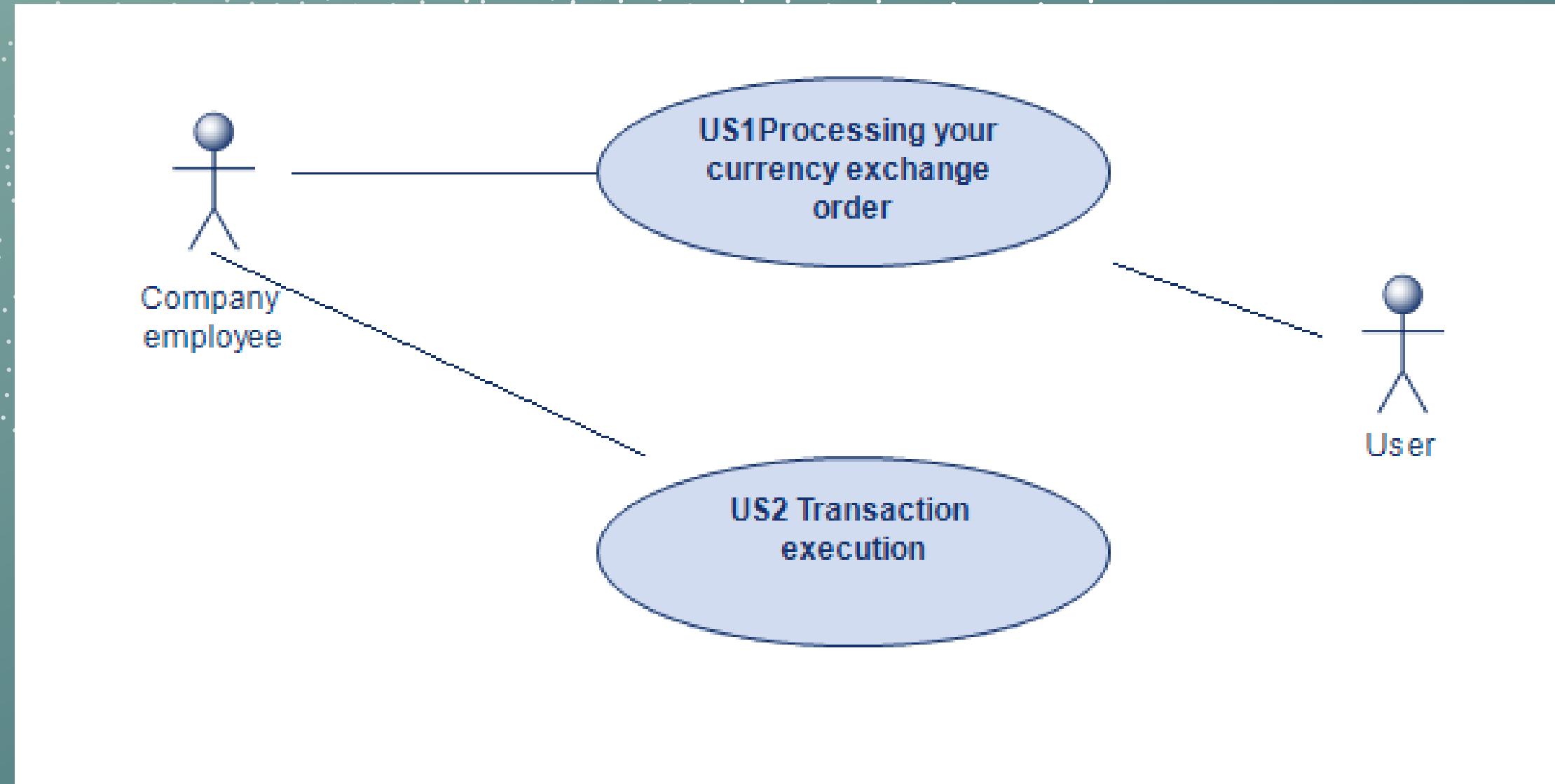
An activity diagram. It shows the system's behavior - you can model various processes carried out by the system

# Context diagram-activity diagram



**Process decomposition diagram.**  
Shows the main sub-processes of the system and the connections between them - models the system's sub-processes using a UML activity diagram

# Process decomposition model



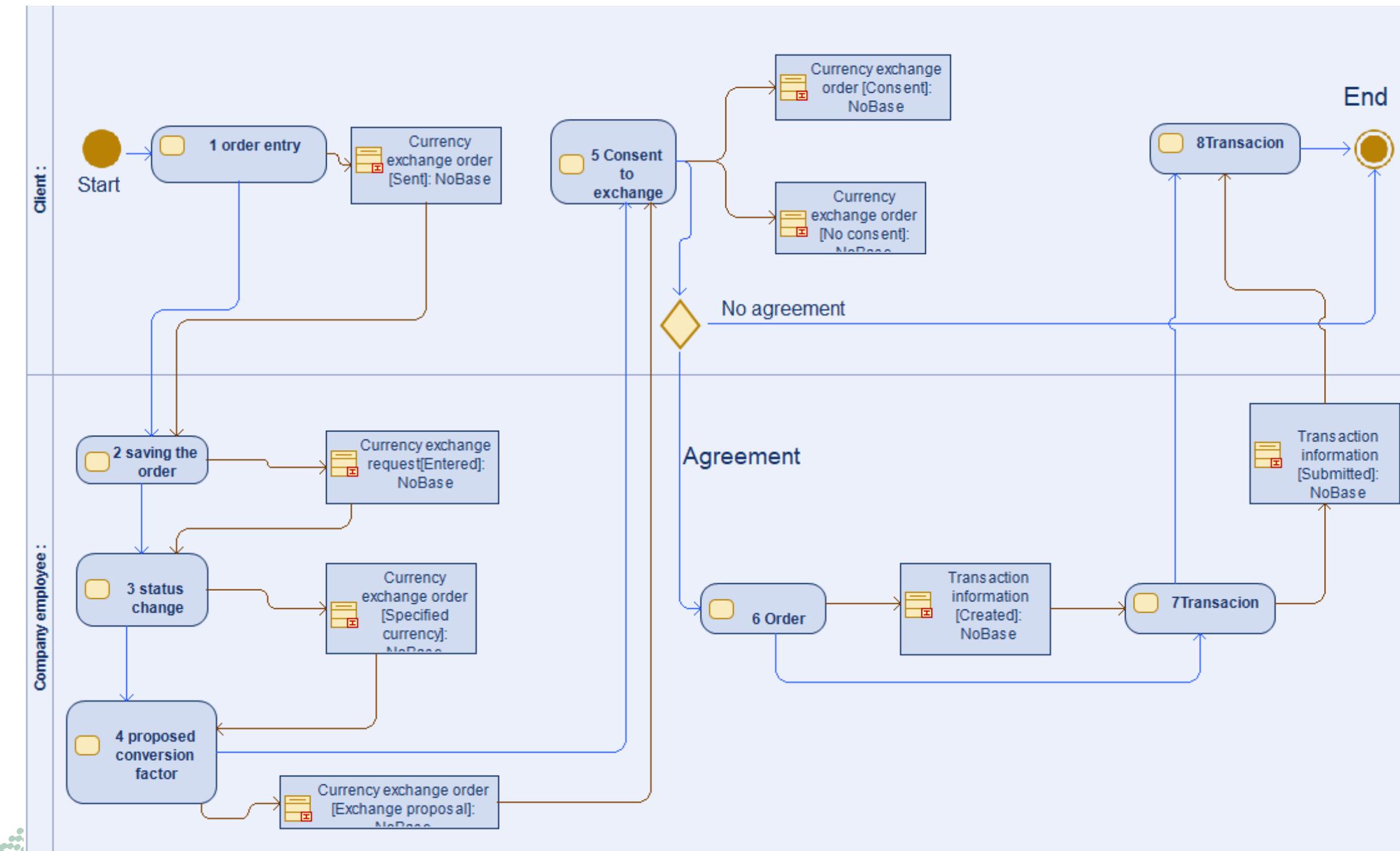
It is a graphical way of representing how users and other external entities will use the system.

# Business use cases



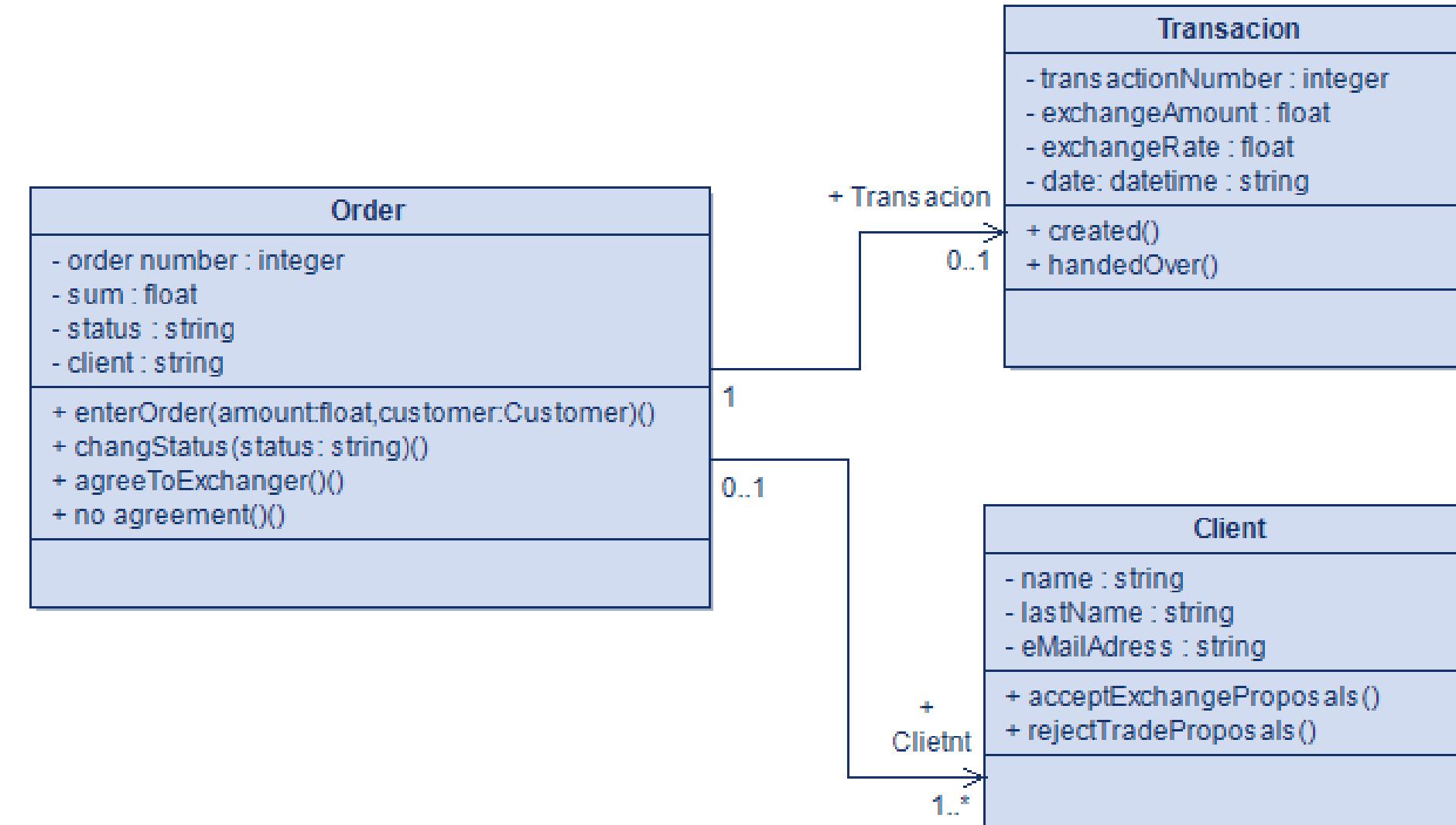
KB

SCIENCE & DESIGN



It presents a sequence of activities and events occurring in a business process. It is a simplified presentation of activities carried out by defined units, leading to the achievement of the assumed goal..

# Implementation diagram of the selected business use case



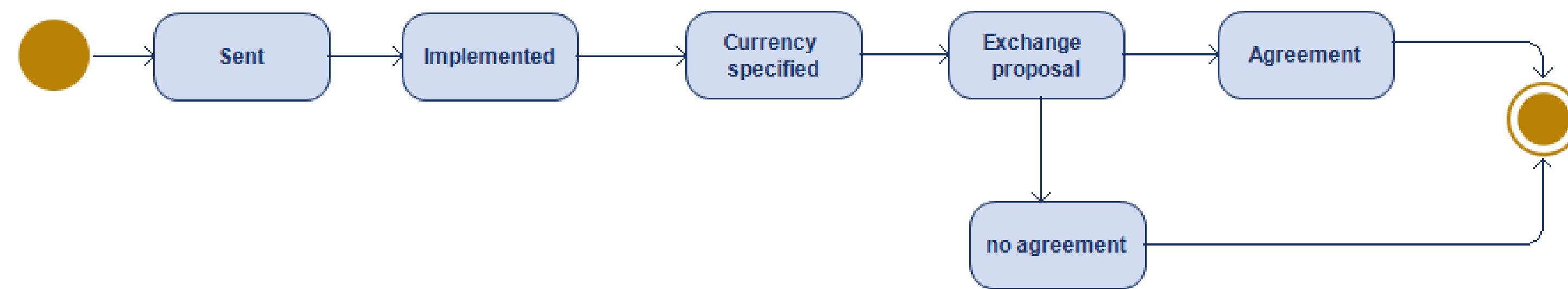
Shows the system structure using main classes and connections between them with a comprehensive number of attributes and operations  
- the system structure is modeled at the system level using a UML class diagram

# Business class diagram



KB  
SCIENCE & DESIGN

## Order



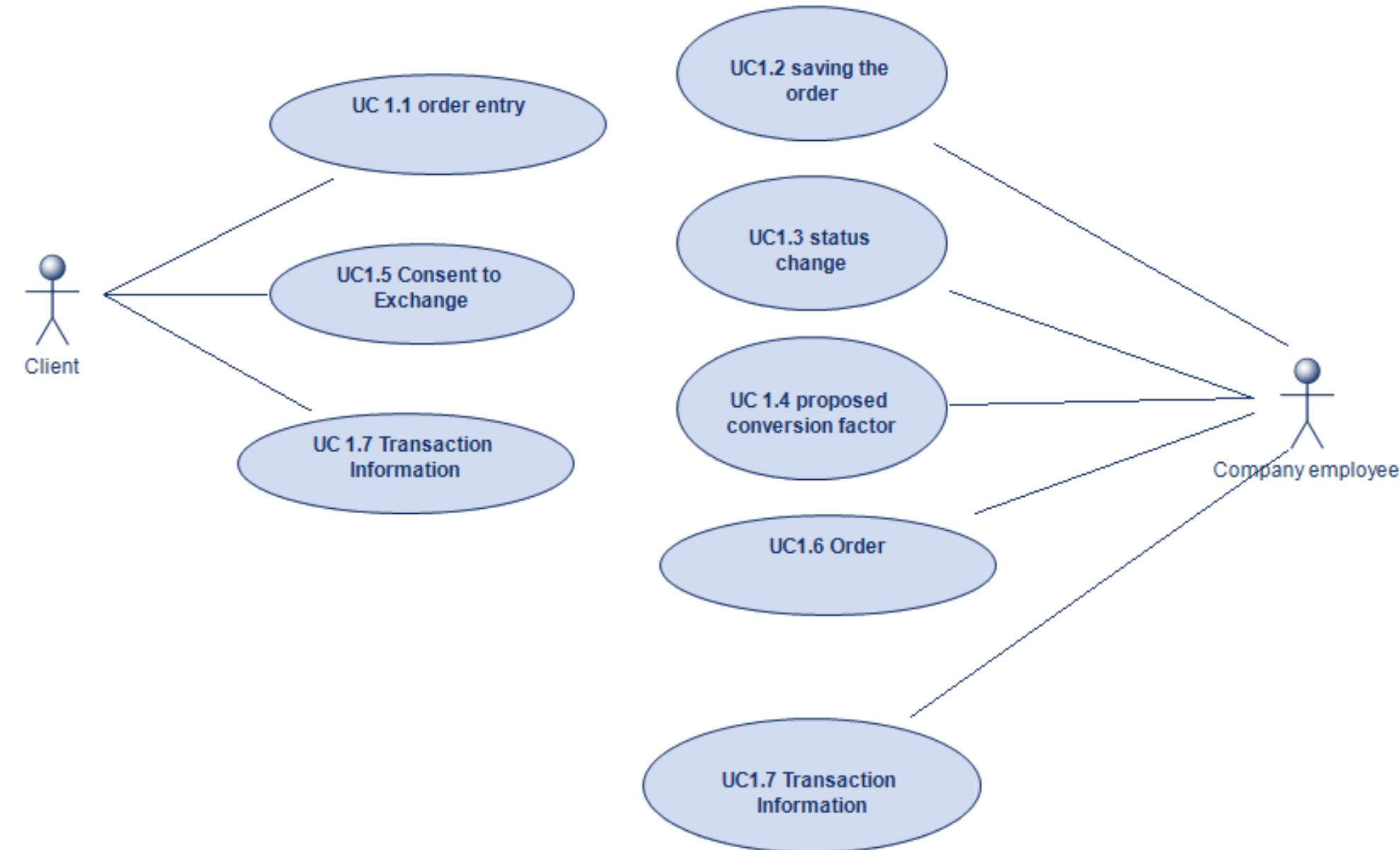
Shows the life cycle of system objects.

# System state machine diagram



KB

SCIENCE & DESIGN



It is a detailed view of the system's functionality. It focuses on the system's internal processes and behaviors. It models specific use cases in which the system performs specific tasks.

# system use case diagram

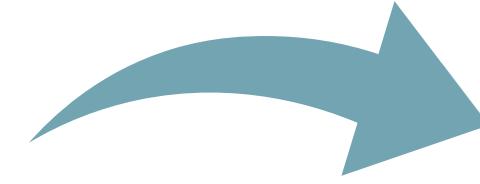
# Process scenario - with at least one alternative course

## process scenario for "Order entry"

1.The user logs in to his account on the currency exchange website.



2. After logging in, the user selects the "Enter order" option.



3.The system displays a currency exchange form where the user can enter the exchange information such as amount, input and output currency.

4.The user enters the required information and confirms the form by pressing the "Confirm order" button.



5.The system checks the correctness of the entered data and available user funds.



6.If the data is correct and there are sufficient funds, the system creates an exchange order and sends the user confirmation of the operation along with the transaction data.

Implementation diagram of the selected system use case

7.The user can now choose to continue (e.g. place another order) or log out of the system

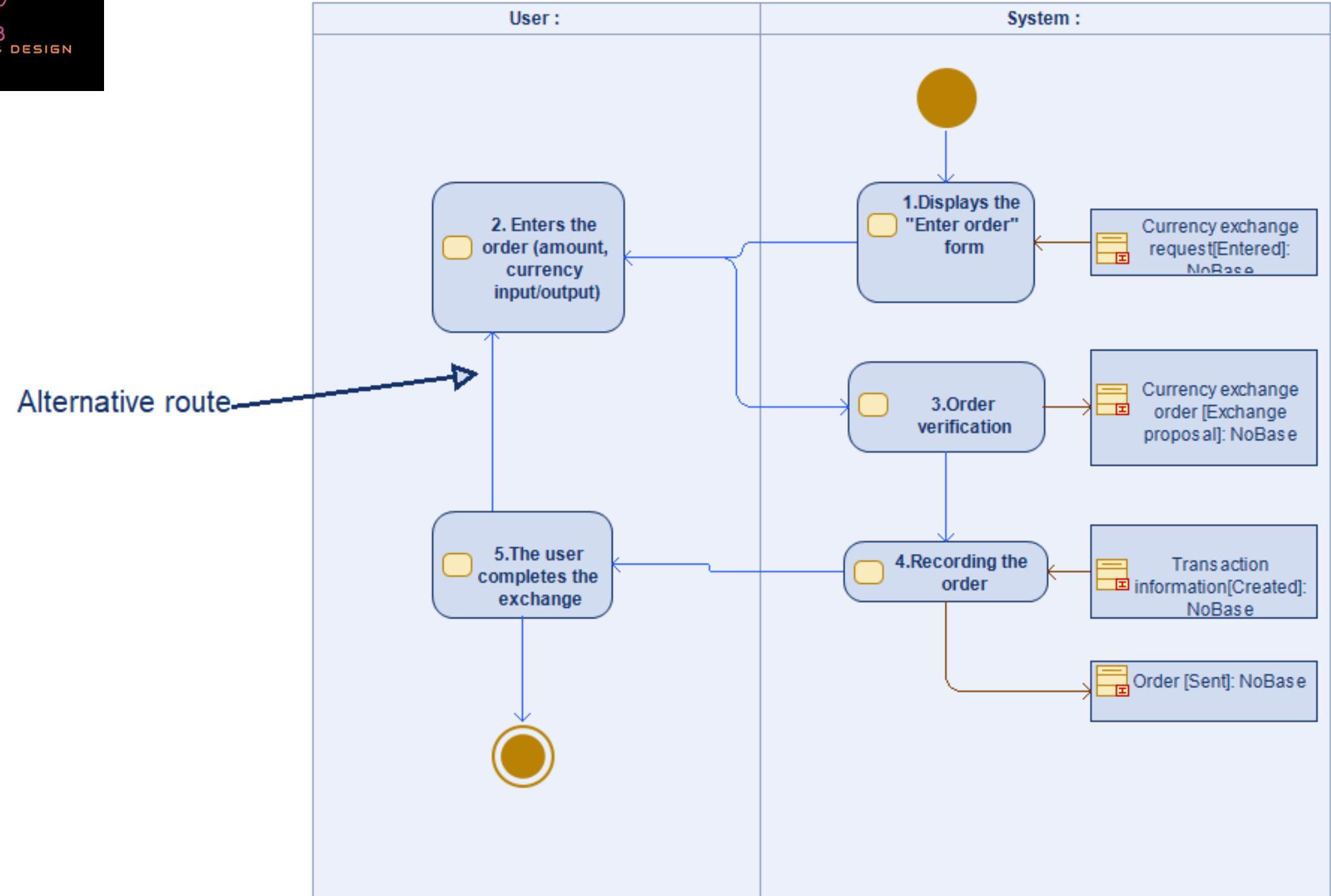
Alternative course





KB

SCIENCE & DESIGN

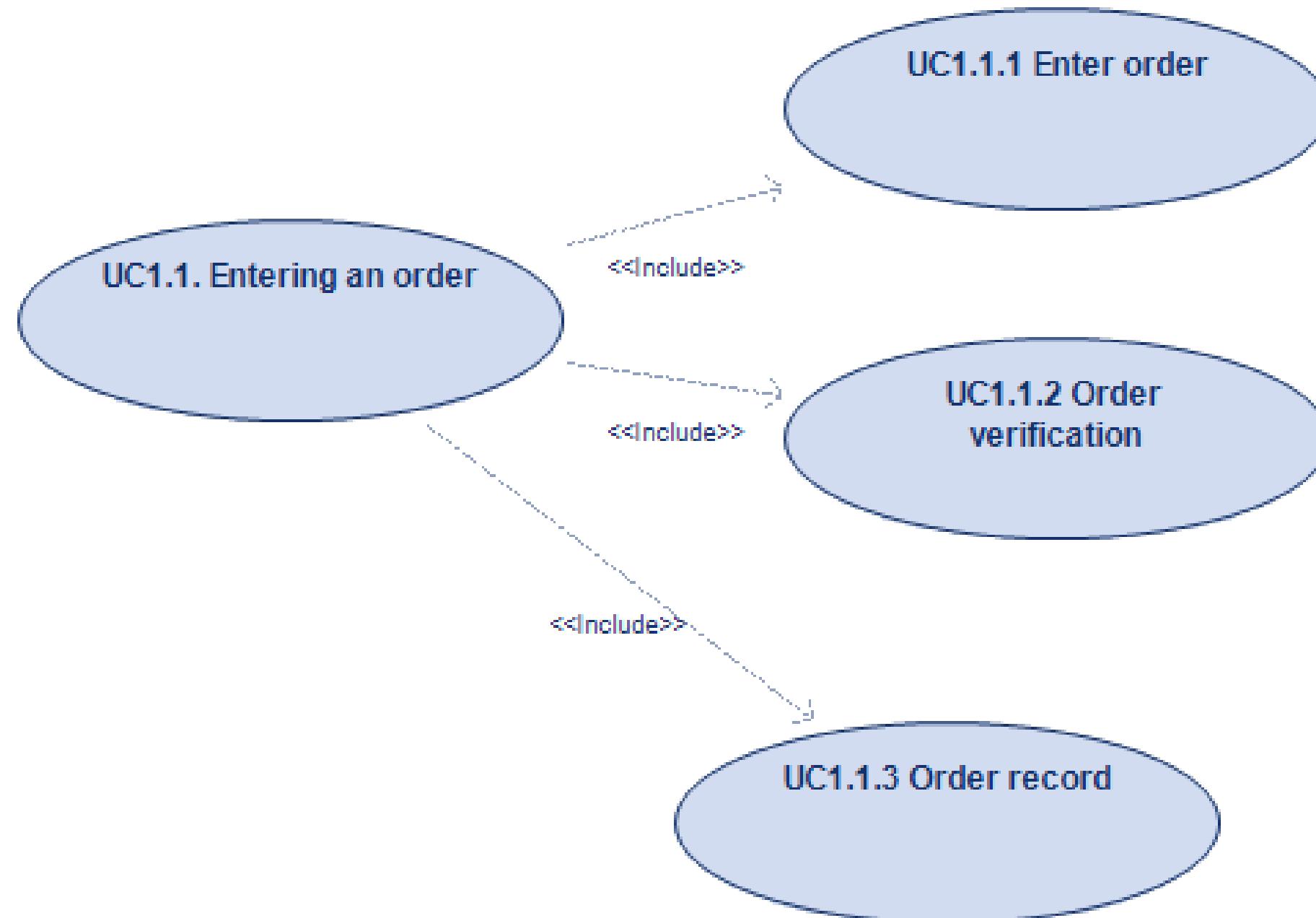


A system use case implementation diagram is a detailed view of the system functionality that models a specific use case. It covers the interactions between actors (users or other entities) and the system within that particular use case

# Implementation diagram of the selected system use case



KB  
SCIENCE & DESIGN



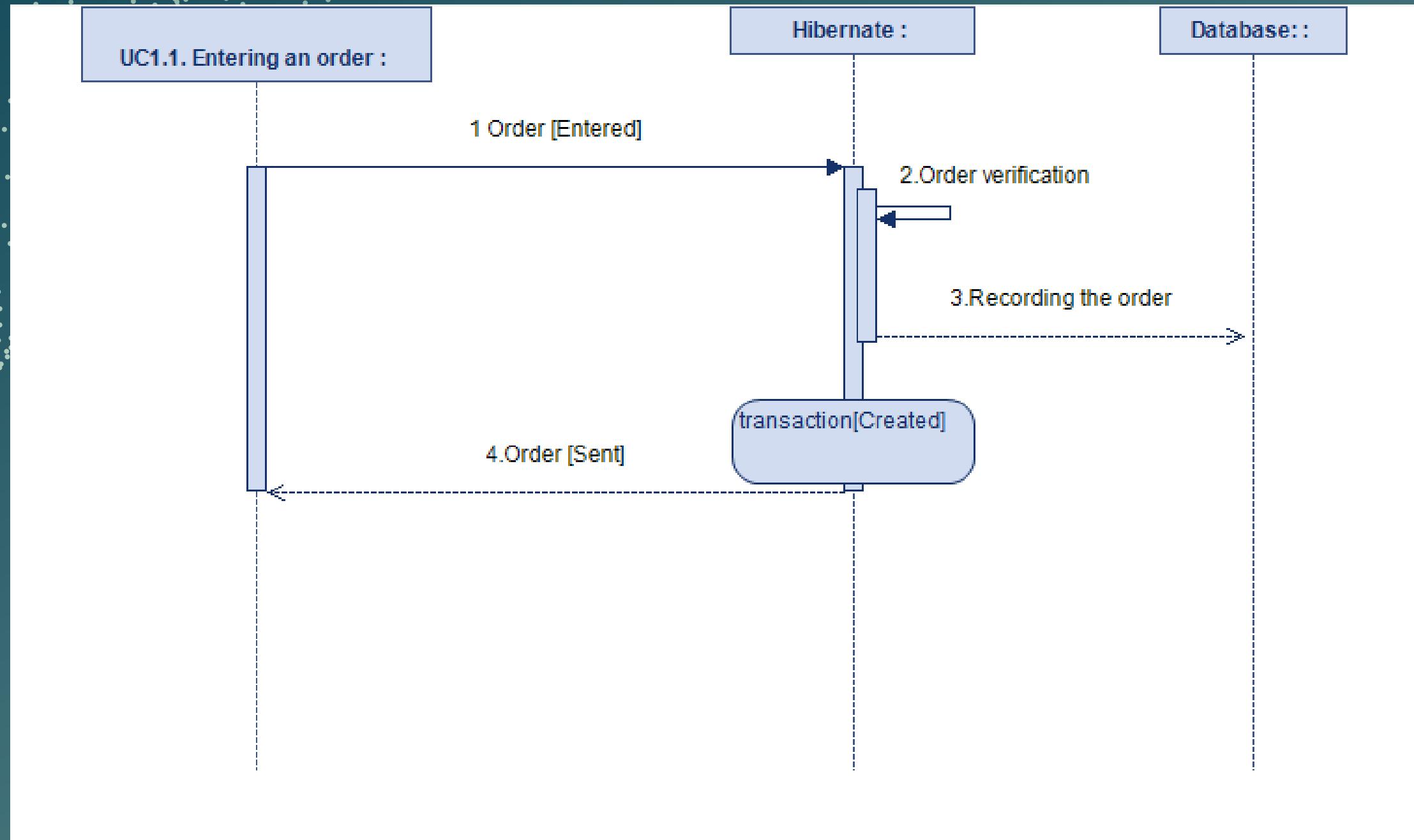
Describes how different elements of a system work together to perform specific functions. Internal use case diagram focuses on detailed modeling of functions inside the system.

# internal use case diagram



KB

SCIENCE & DESIGN

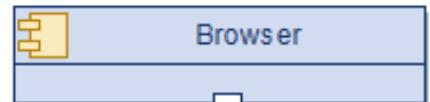


Describes the exchange of information between system objects. The exchange of information in these diagrams is presented chronologically, usually in the form of sending messages between objects (a message can also be sent to the object that sent it). Typically, sequence diagrams are used to model the behavior of system objects (components).

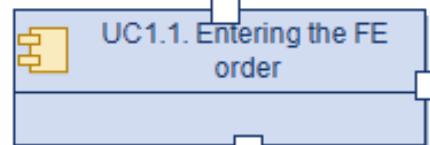
# Sequence diagram



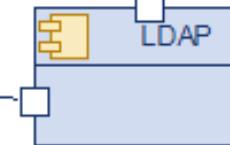
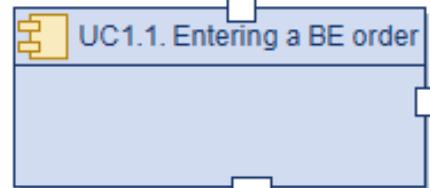
## Client Layer



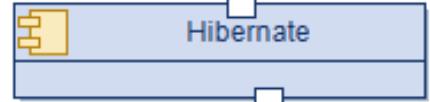
## Presentation Layer



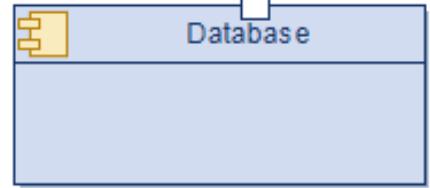
## Business Layer



## Integration Layer



## Data Layer



# Component diagram

Describes the structure of the IT system software. It describes the individual elements of the system software and the interfaces between them, which enable the transfer of information between components in the form of providing services and requesting services. Component diagrams are typically used to model the implementation perspective of an IT system. In most cases, individual components correspond to system objects identified in the logical perspective (class diagram, object diagram).