

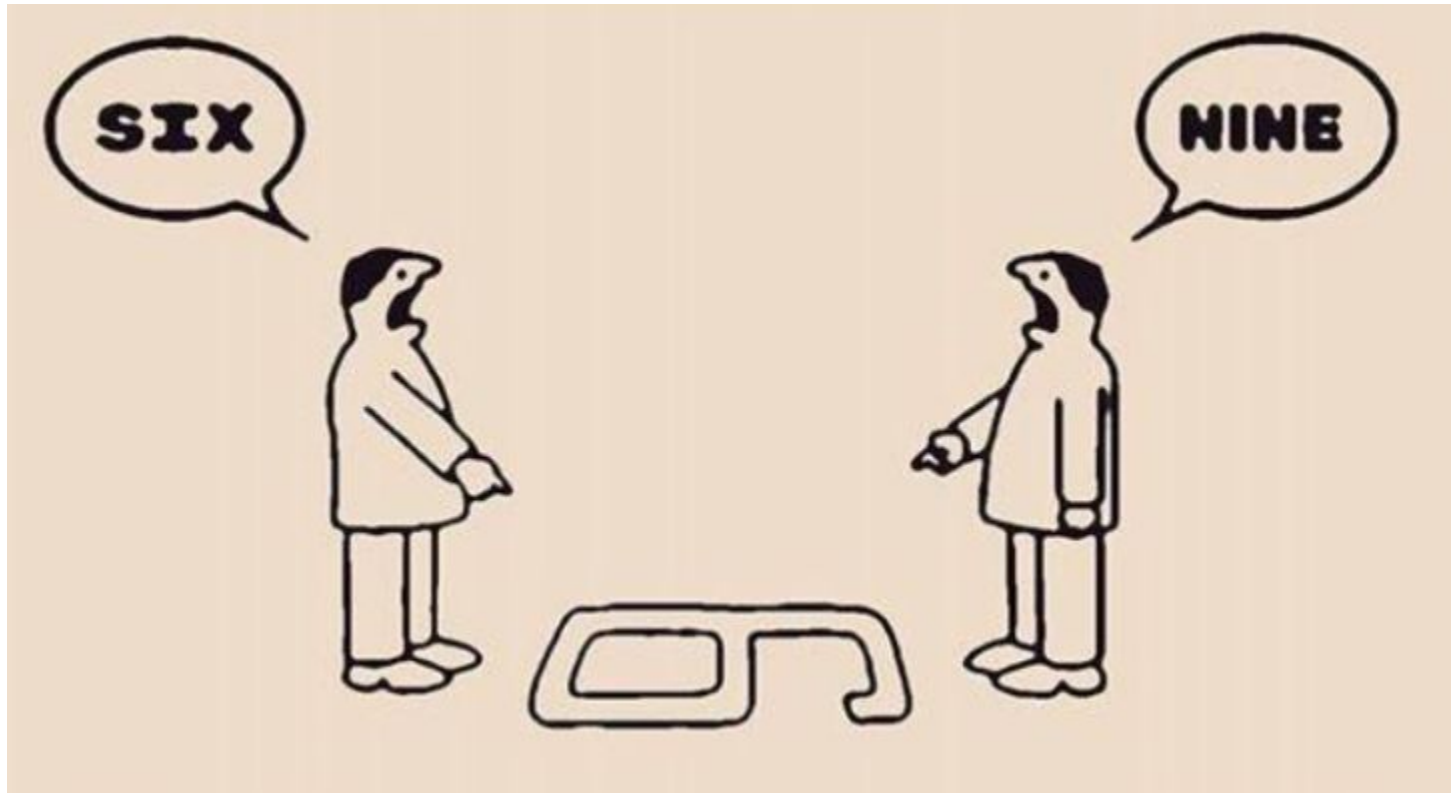
UML101

By Fiștic Cristofor

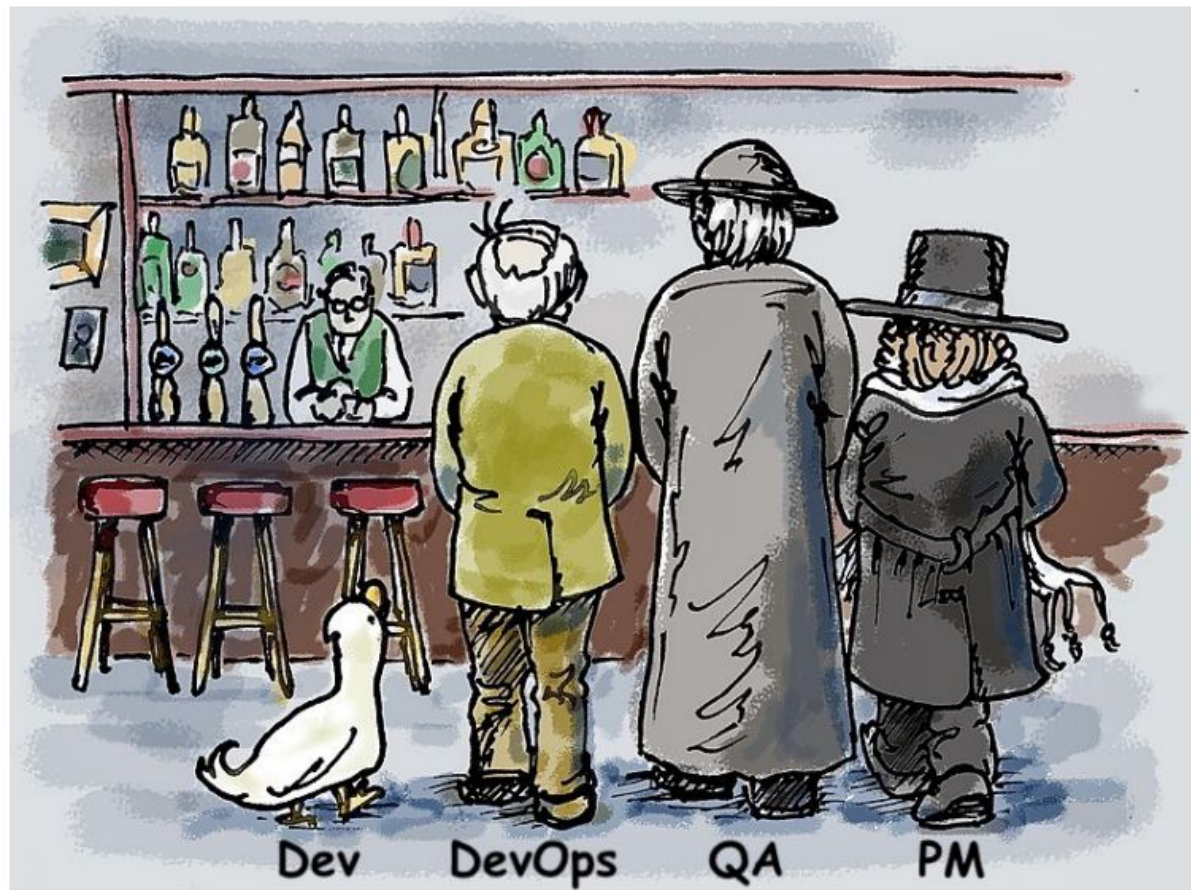
Table of context:

- Introduction
 - Problem
 - Solution
- Diagrams
 - Structural
 - Use Case
 - Behavioral
 - Sequence
 - Activity
- Tools
- References

Problem



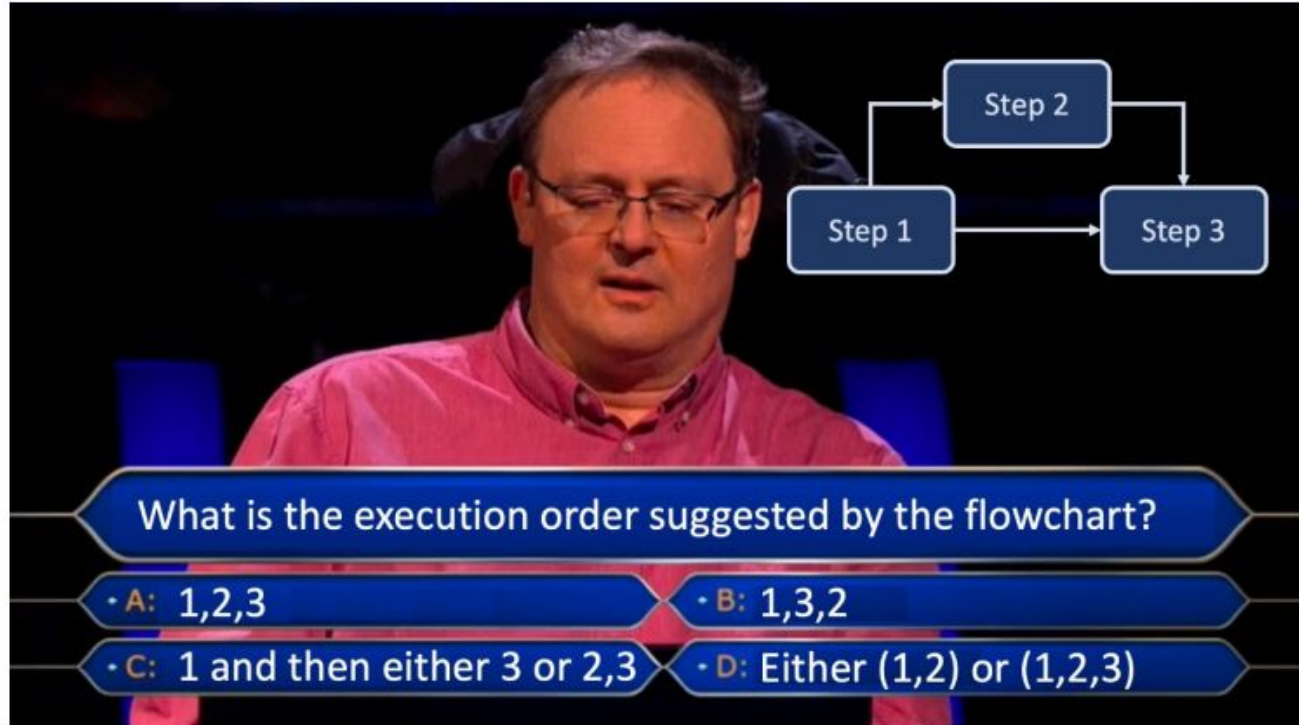
Problem



Problem



Problem - Ambiguous Diagrams

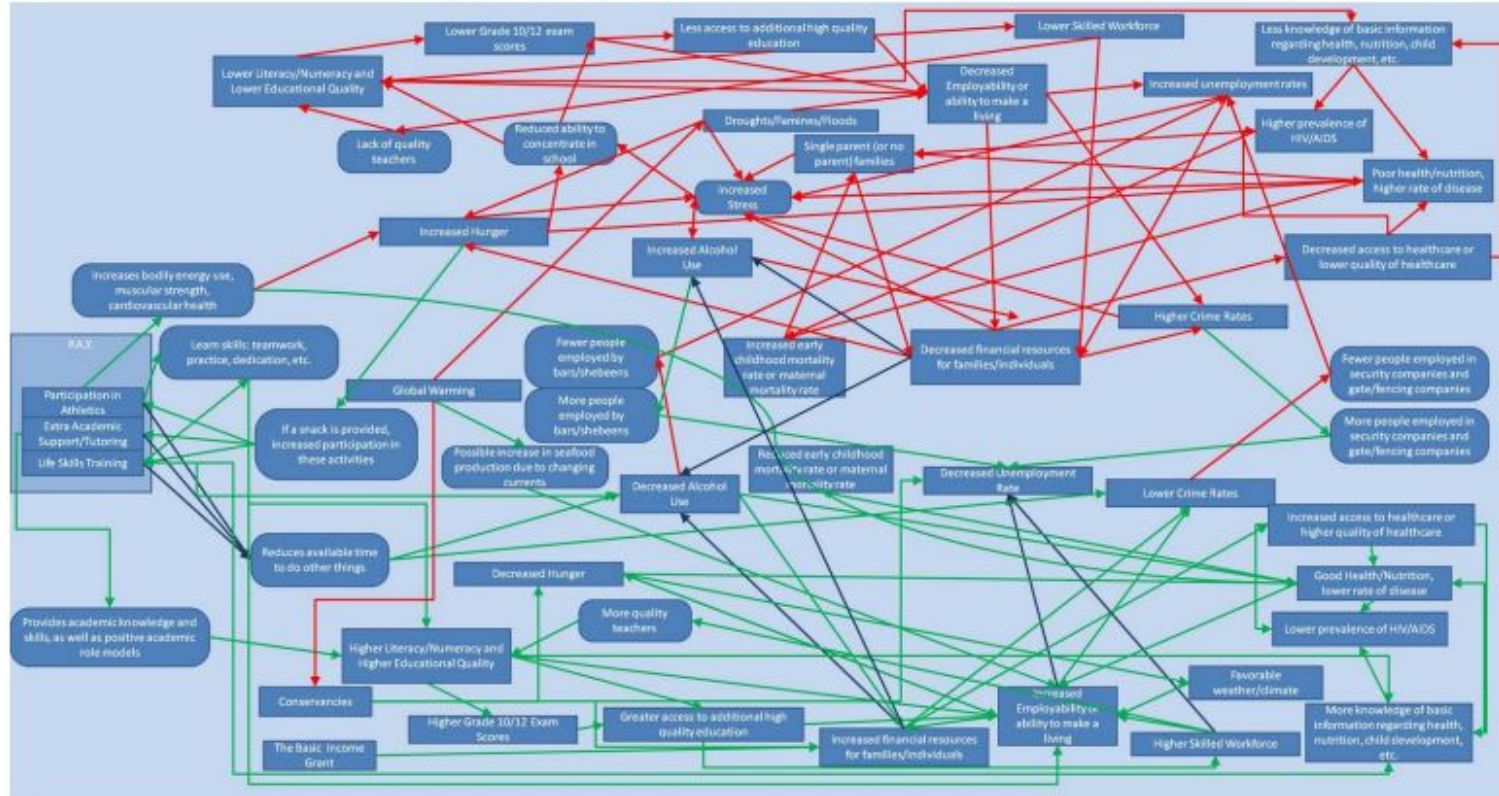


```
graph LR; S1[Step 1] --> S2[Step 2]; S1 --> S3[Step 3]; S2 --> S3;
```

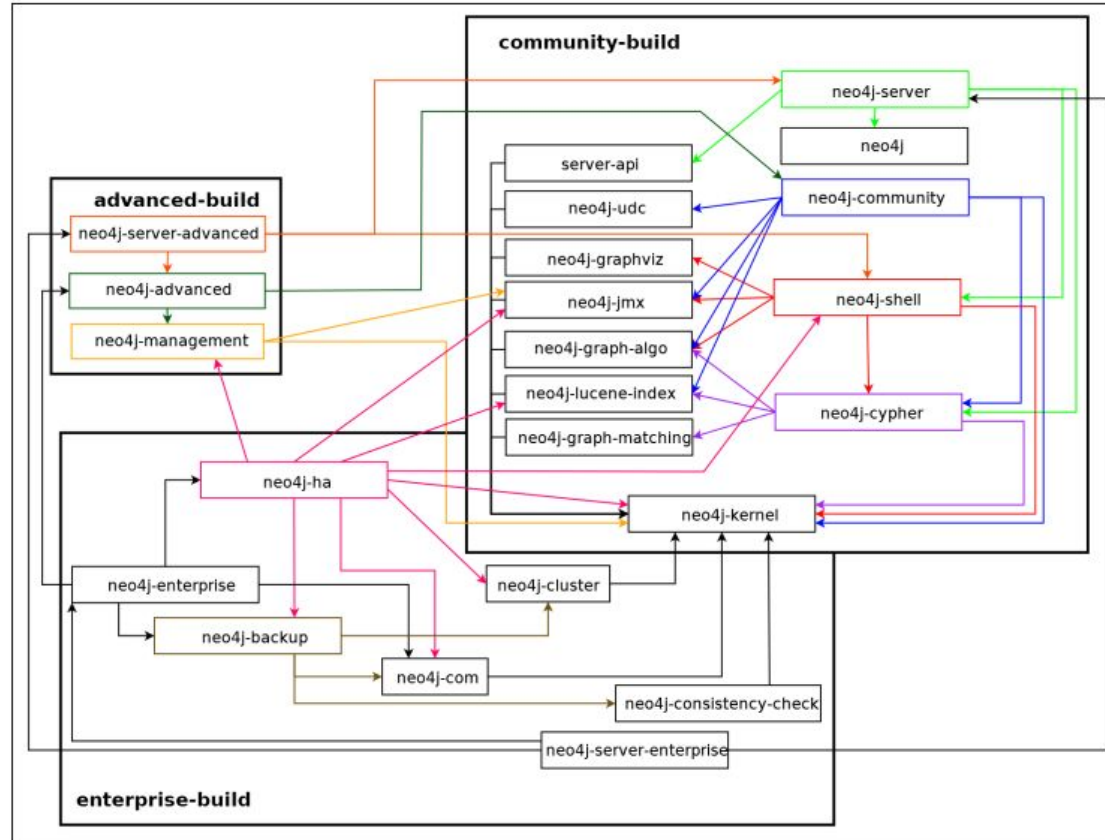
What is the execution order suggested by the flowchart?

- A: 1,2,3
- B: 1,3,2
- C: 1 and then either 3 or 2,3
- D: Either (1,2) or (1,2,3)

Problem - Chaos



Problem - Chaos



Solution

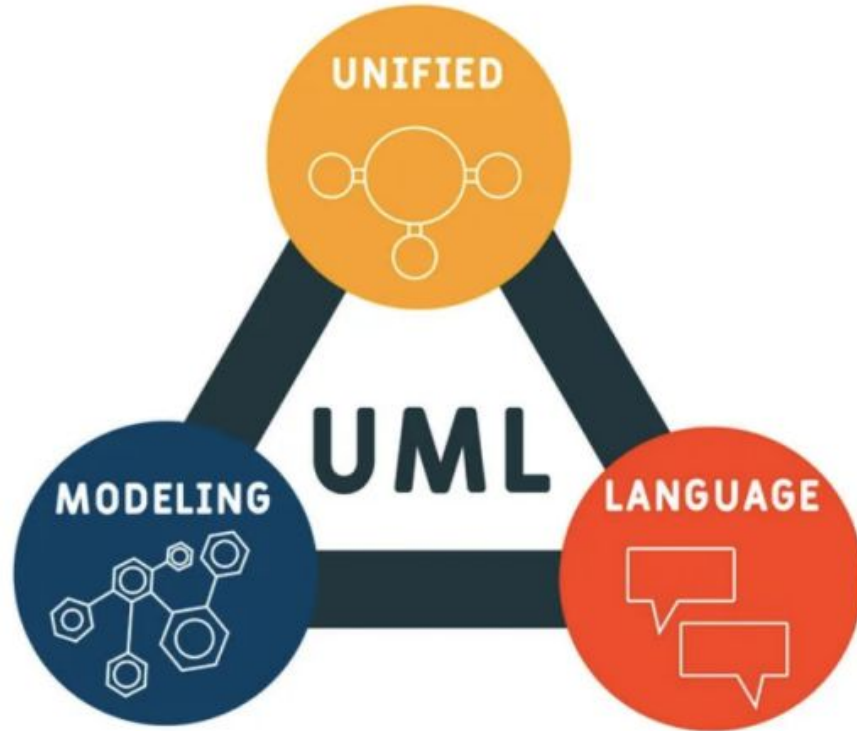


Diagram - Use Case

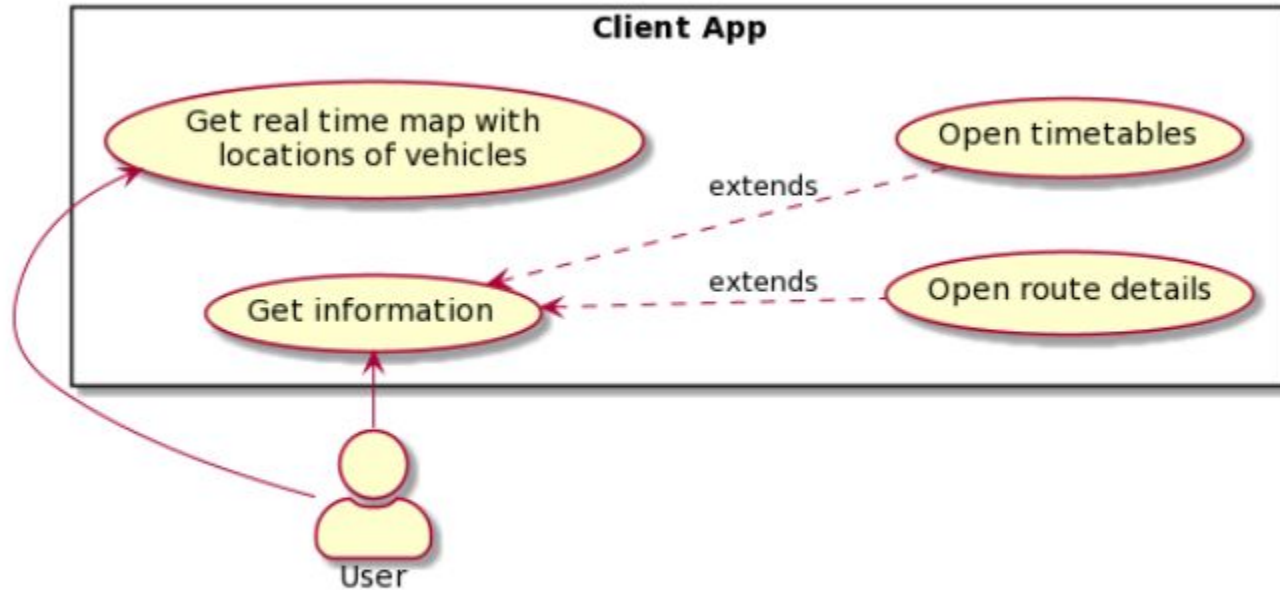


Diagram - Use Case

- Include the most relevant use case (domain specific ones).
- Keep them simple and coherent.
- Different diagrams for different
 - types of actors,
 - subsystems,
 - perspectives

Diagram - Activity

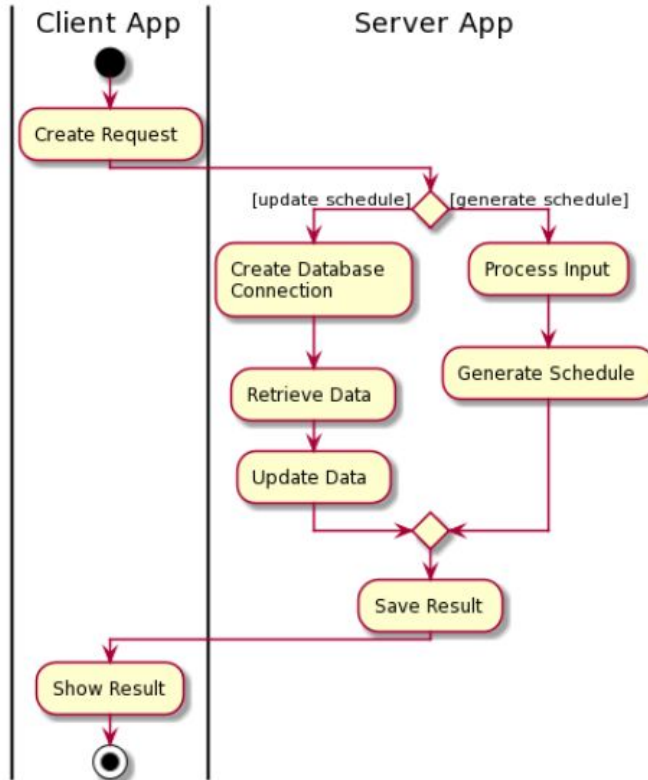


Diagram - Activity

- You can group state transitions into states
- In case you do that provide start and end states

Diagram - Sequence

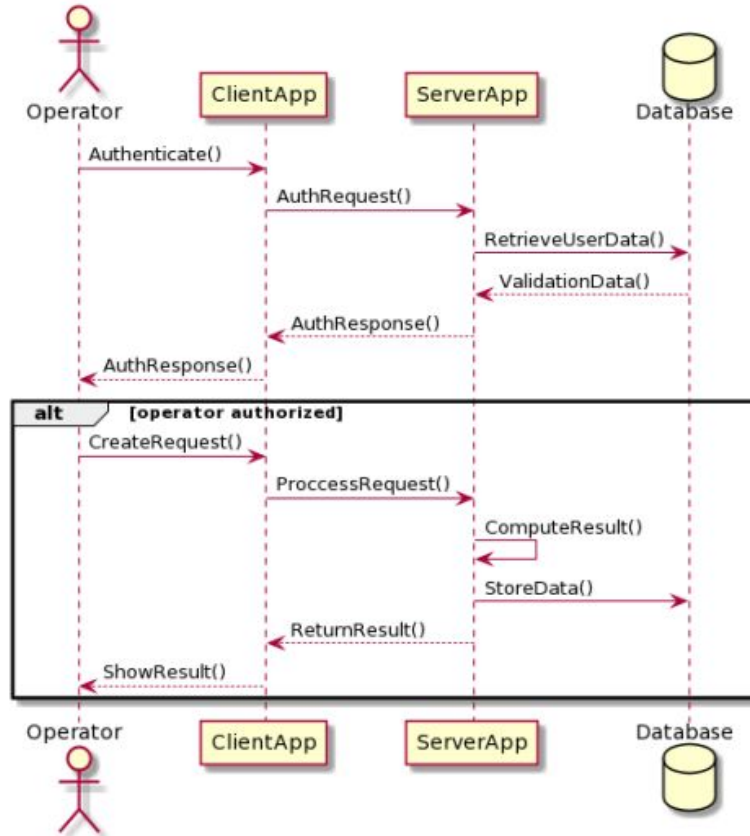


Diagram - Sequence

- All the requests should be ended by responses or destruction occurrences.
- Synchronous requests should be followed by a response or an asynchronous request.
- Divide the sequences using fragments.

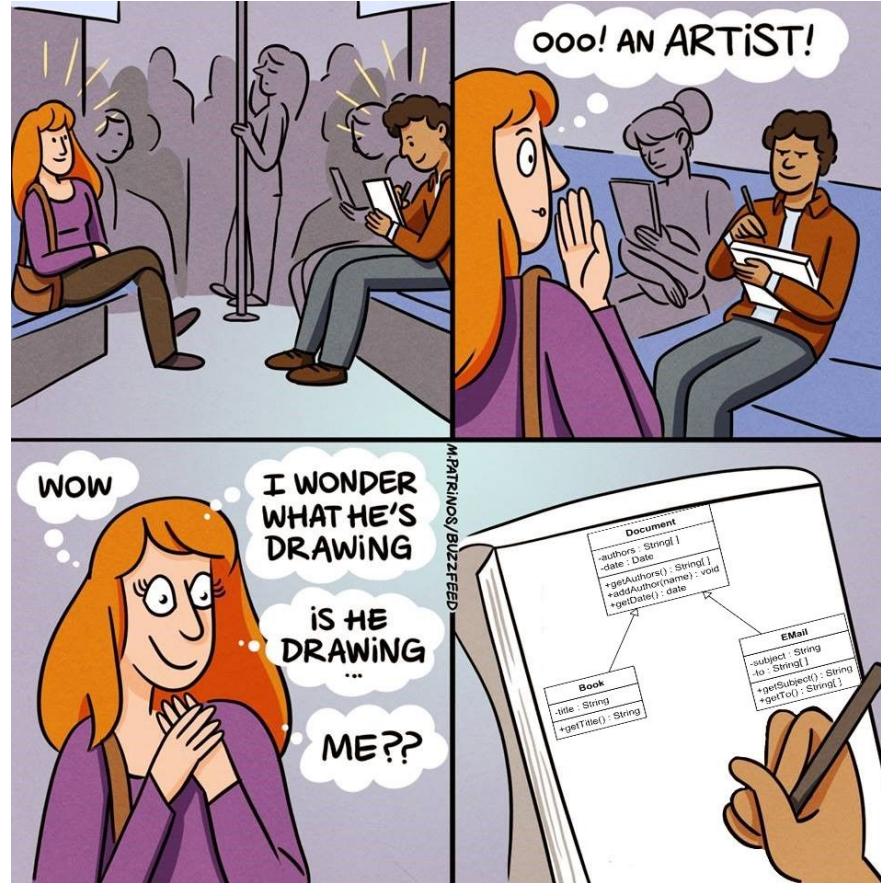
UML - Strengths:

- Multiple perspectives of a system
- Non Ambiguous (if used correctly)
- Varies from simple/intuitive to complex/technical

Tools:

- [Draw.io](#)
- [PlantUML](#)
- [Lucidchart](#)

After lesson



References:

- [PlantUML](#)
- [UML web site](#)
- [Activity diagram](#)
- [Use case diagram](#)
- [Sequence diagram](#)