INSTITUTO SUPERIOR
DE TECNOLOGIAS
AVANÇADAS DE LISBOA

Software Engineering

João Caldeira

Invited Professor

Email. joaocarlos.caldeira@my.istec.pt

Mob. +351 917769544

November 30, 2022

ISTEC INSTITUTO SUPERIOR DE TECNOLOGIAS AVANÇADAS DE LISBOA

Software Design

November 30, 2022

- **Definition**

1. It’s the process of developing abstract models of a system, with each model presenting a different view or perspective of that system

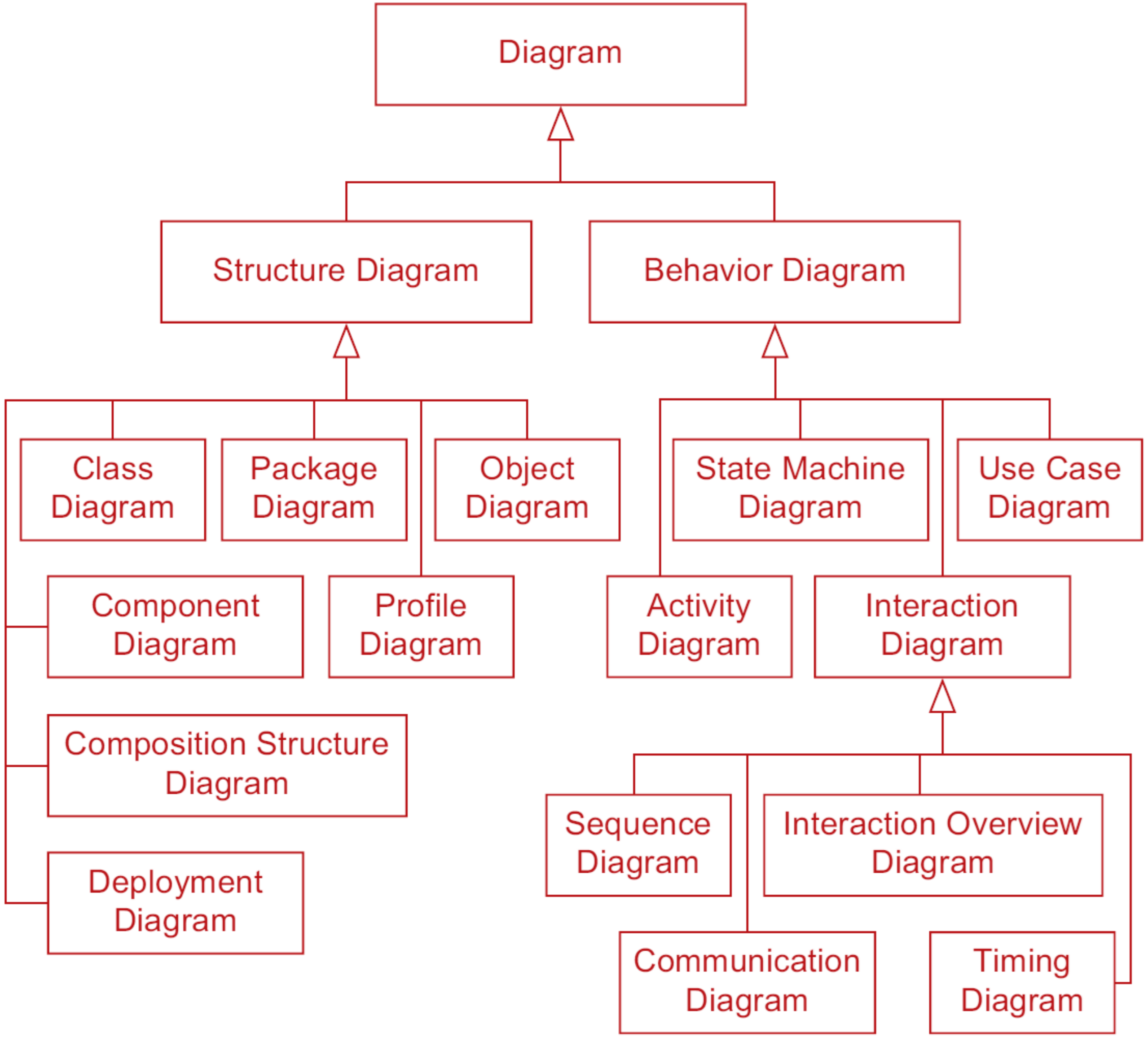
2. Usually means representing a system using some kind of **graphical notation** based on diagram types in the **Unified Modeling Language (UML)**

3. It is also possible to develop formal (mathematical) models of a system, usually as a detailed system specification

Software Design

System Modeling

- **UML Diagrams**



- # Diagram Types

1. **Activity diagrams.** Show the activities involved in a process or in data processing
2. **Use case diagrams.** Show the interactions between a system and its environment
3. **Sequence diagrams.** Show interactions between actors and the system and between system components
4. **Class diagrams.** Show the object classes in the system and the associations between these classes
5. **State diagrams.** Show how the system reacts to internal/external events

Software Design

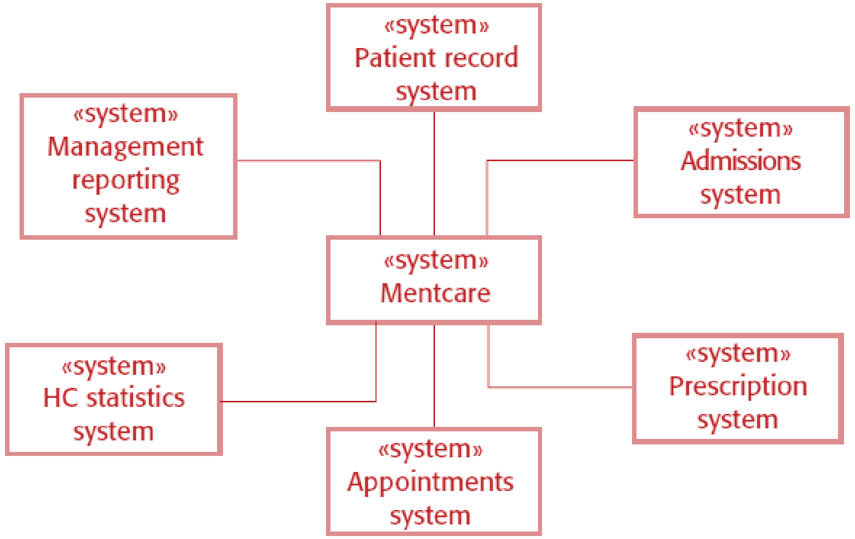
Context Models

- **Definition**

1. A model is an abstract view of a system that deliberately ignores some system details

2. Complementary system models can be developed to show the system's context, interactions, structure, and behavior

3. **Context models show how a system that is being modeled is positioned in an environment with other systems and processes**. They help define the boundaries of the system to be developed



INSTITUTO SUPERIOR DE TECNOLOGIAS AVANÇADAS DE LISBOA

Software Design

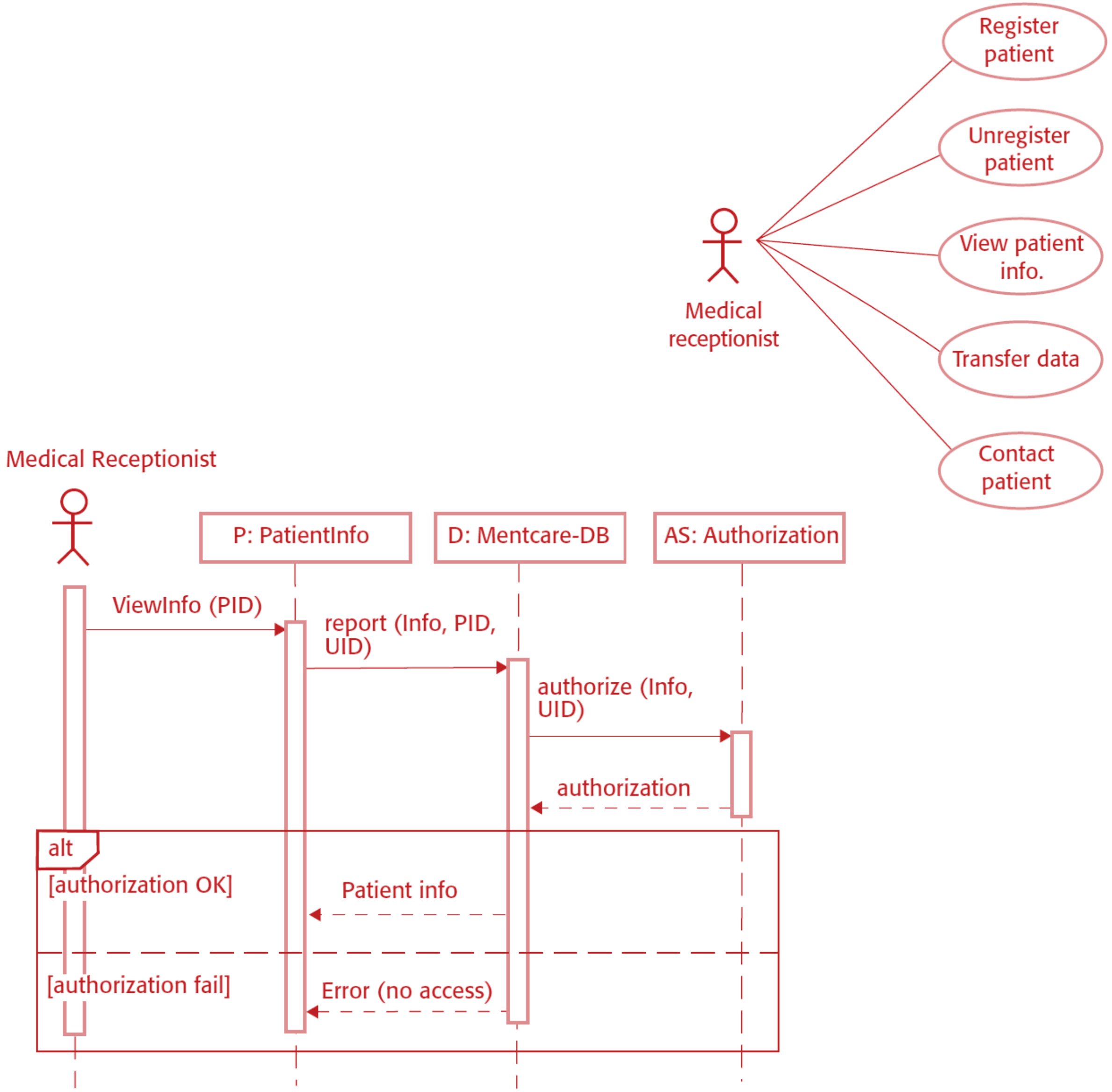
Interaction Models

- **Definition**

1. **Use case** diagrams and sequence diagrams are used to **describe the interactions between users and systems in the system being designed**

2. **Use cases** describe interactions between a **system and external actors**

3. **Sequence diagrams** add more information to these by showing **interactions between system objects**



Software Design

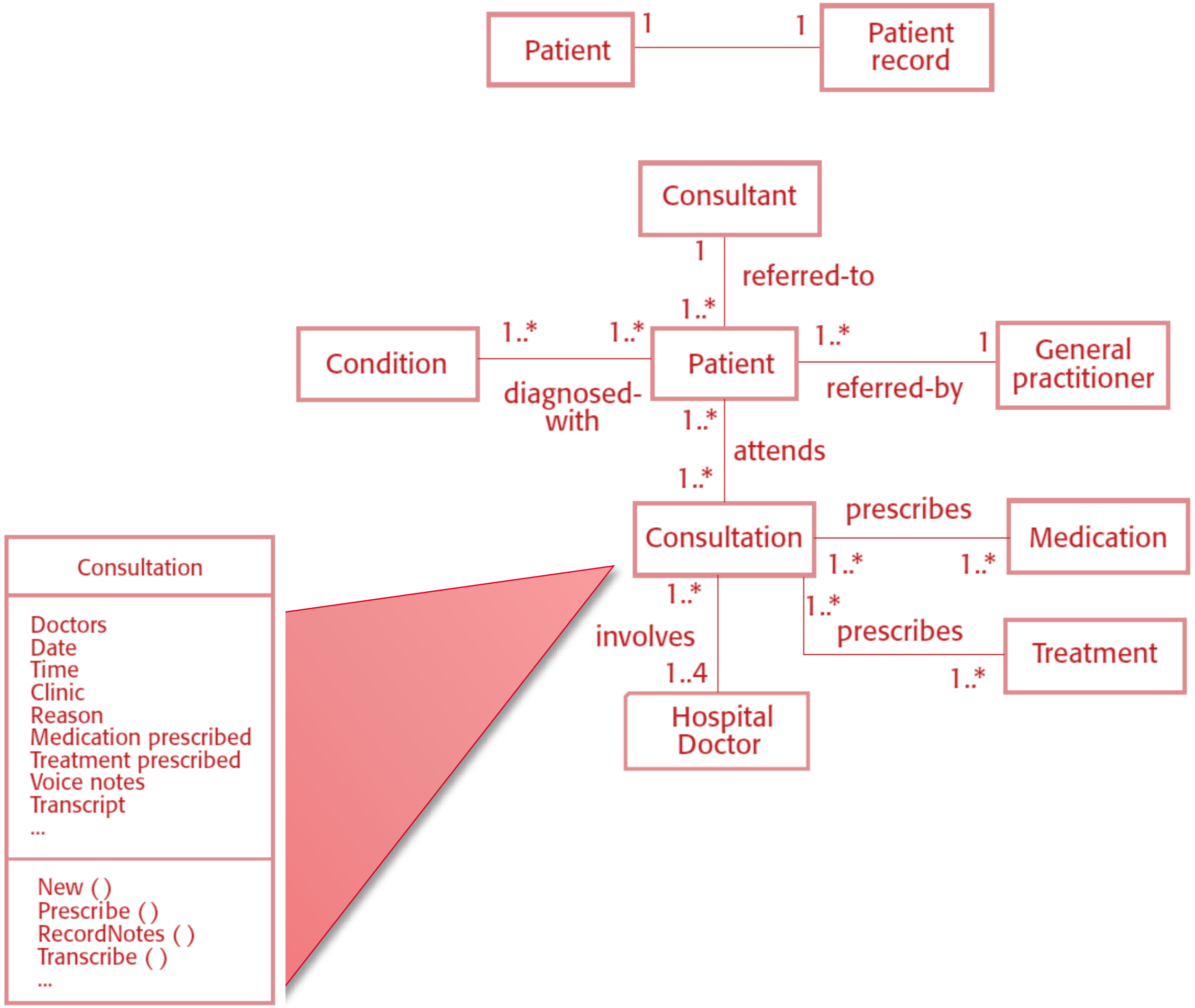
Structural Models

- **Classes & Associations**

1. Structural models show the organization and architecture of a system

2. Class diagrams are used to define the static structure of classes in a system and their associations

3. The **operations** (called **methods** in **Java and other OO programming languages**) associated with the object class are in the lower section

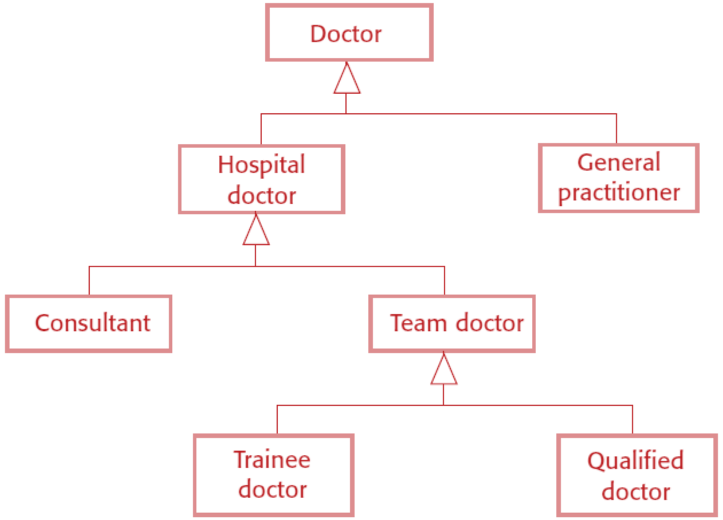


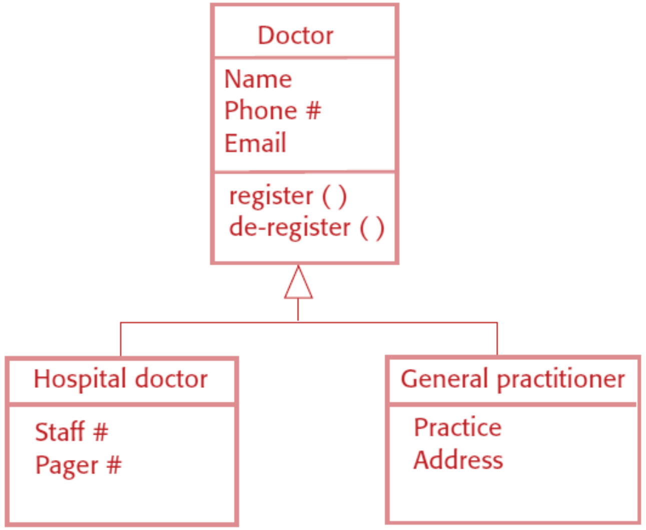
Software Design

Structural Models

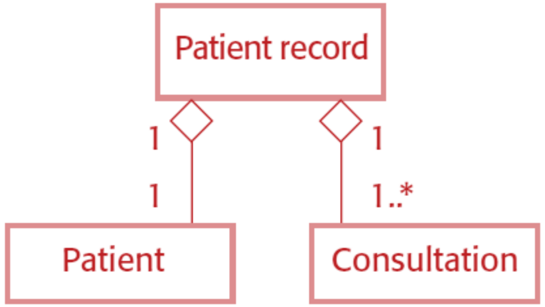
- **Generalization Hierarchy**

  1. The name of the object class is in the top section
  2. The middle section contains the attributes and optionally their types





- **Aggregation**



Software Design

Behavioral Models

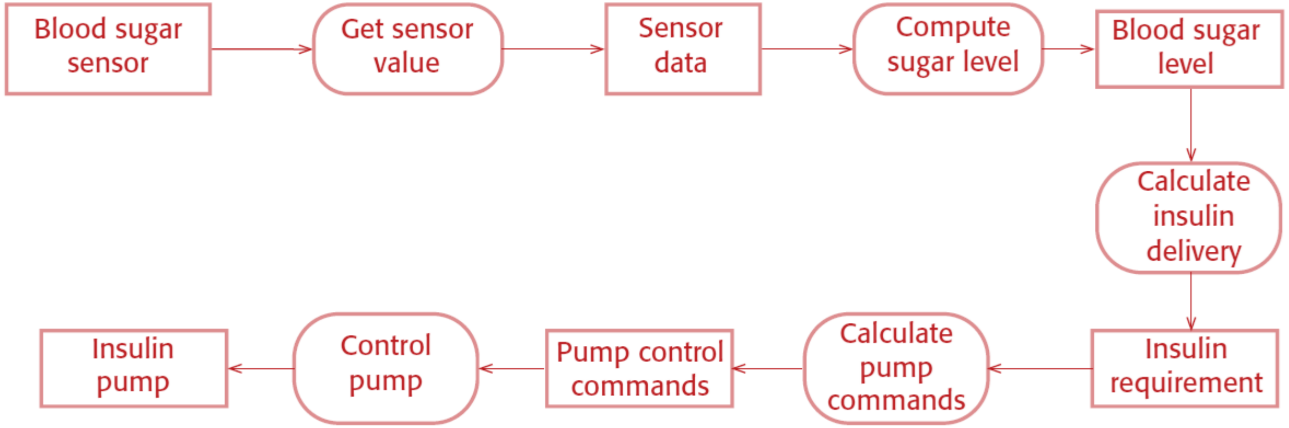
- **Data-Driven Modeling**

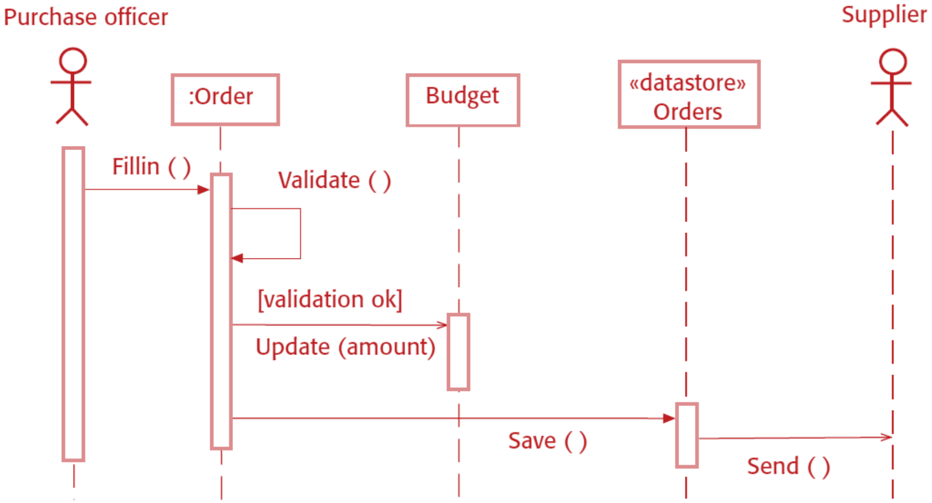
1. Behavioral models are used to describe the dynamic behavior of an executing system

2. Can be modeled from the perspective of the data processed by the system or by the events that stimulate responses from a system

- **Activity Modeling**

- **Sequence Modelling**



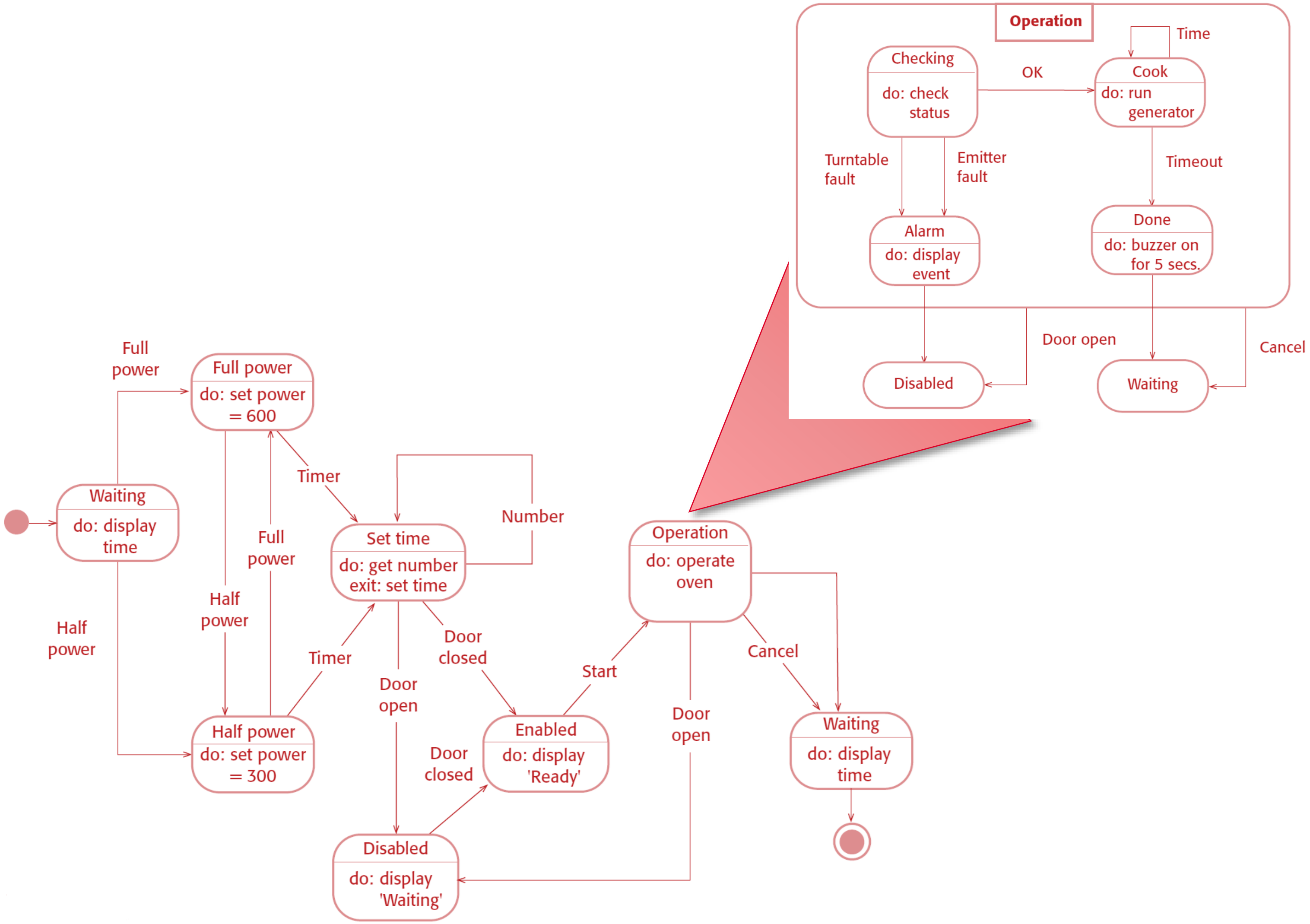


Software Design

Behavioral Models

- **Event-Driven Modeling**

1. **State Diagrams** are used to model a system's behavior in response to internal or external events



Copyright @ 2022