Requirements engineering

# Role of requirements

Requirements must define:

**WHAT** the system should do

**NOT HOW** it should do it

Requirements must describe:

**WHAT** the product must do

Or a **QUALITY** it must have

A requirement is:

A **CONDITION** or **QUALITY** that must be met or possessed by the system

**Example:** consider a software system which provides a booking functionality for theatre tickets

**Problem:** what the system should do in order to enable users to book a theatre ticket?

**Analysis:** what would enable users to book? Which data are involved? Storage? Registration? Booking a regular seat or box seat? Availability of seats?

# Requirements elicitation

The process of finding and formulating requirements from the **stakeholders**, **documents**, and **other sources**.

Involves **careful study** of the **product domain** and **stakeholder needs**.

Major activities:

* Requirements discovery
* Requirements classification and organisation
* Requirements prioritization
* Requirements documentation

# Onion model

**System boundary**

Actors that directly interact with the software

# Requirement Classification and Organisation

After getting the user’s requirements, we must develop the system requirements which is much more detailed. The system requirements are gathered and determined via further requirement elicitation from the client and a literature review to how other similar product work.

# Use Case Notation

Represent only the actors that benefit from the system

Don’t use “Login” for use case; instead, use something like “Login user”. Don’t just use a very for the use case

# Requirements Modelling

**Note do the domain model in Astah and everything else in StarUML**