Components –

Boundary – Scope of system. Finite functionality. Defines limits of system.

Actors – Individuals involved with the system defined according to their roles e.g. bank actor = customer

Use cases – Specific roles played by actors. Representation of distinct business functionality

Associations – Between and among actors and use cases

Good for

Identify and organise system requirements

End user and non-technical stakeholders are considered and can understand the model. Feedback obtained early on

Use for other aspects of software development – cost estimates, planning, testing

Need to identify rare/exceptional scenarios – comprehensive requirements

Should not be used for

Represent exception/error behaviours

Represent sequence of steps that must be performed to complete a task

Individual classes – breaking down and reassembling