Glossary

Activity diagram  A UML diagram used to describe workflows, used in a variety of modelling roles from business process modelling through to the detailed description of operations.

Actor  An external entity (person or system) that interacts with a computer system.

Alternative path  A normally used alternative route through a business process or use case.

Application architecture  The large-scale structure of the application software.

Architecture  The large-scale design of the structure of a system.

Association  A logical link between two classes, indicating that instances at either side of the association may communicate with the other.

Atomicity  The property of a database transaction that it must either succeed or fail without side-effects.

Business model  A description of a business operation, using models such as a process map and activity diagrams, with other elements such as a stakeholder analysis.

Class  A collection of objects with identical structure.

Class diagram  A UML diagram that shows classes and their associations, indicating the static structure of a system.

Collaboration diagram  A UML diagram that shows how instances of objects interact to provide functionality, usually in the realization of a use case.

Component  A unit of software that implements a logical set of functionality, usually implemented as a set of objects.

Component diagram  A UML diagram that shows the interdependency of components in a system.

CORBA  Common Object Request Broker Architecture – a set of standards defining the interaction of objects in a distributed environment.

Cost–benefit model  A model of cash flows intended to describe how costs and benefits change over time during a project. This is usually handled in a spreadsheet.

Deliverable  A deliverable is a key output of a project. It may be a final deliverable, such as a module of a system to be deployed,
or an intermediate deliverable, such as a use case definition or a design.

**Design**
The process of converting the requirements expressed by systems analysis into a set of objects and components that can be implemented in a computer system.

**DSDM**

**Event**
An occurrence (either inside a system or outside) that triggers some activity.

**Exception**
An unusual circumstance in a business process, use case or operation on a class that requires special action.

**Extreme Programming**
A method that avoids the use of explicit modelling in the design and construction phase, normally known as a lightweight method.

**First normal form**
A table is said to be in first normal form if every record has a fixed number of simple fields.

**Foreign key**
When the key to another table is stored as a value or set of values in another table for the purposes of combining the data in the two tables, this is known as a foreign key.

**Functional requirement**
A requirement of a system that states what the system should do, usually expressed as a use case (or part of a use case).

**Goal**
An objective of a stakeholder, or of a business process or use case.

**IIOP**
Internet Inter-Orb Protocol – a protocol for allowing ORBs to communicate to exchange requests and responses between objects.

**Iterative development**
The development of software in stages, resulting in multiple releases on increasingly functional software.

**Join**
A way of combining two tables together so that related information can be viewed together.

**Key**
A value or set of values that uniquely identify a record in a database table.

**Methodology**
Strictly, the study of methods for the development of software. More commonly used to mean a set of methods and notations used to develop a system.

**Milestone**
A milestone is a stage of a project where some set of agreed deliverables is provided.

**Moscow**
This is a prioritization convention that breaks actions or goals into those that Must be achieved for a task to be successful, those that Should be achieved by the task, those that Could be achieved by the task if there is time and resource available, and those that Won’t be considered by the task.

**Multiplicity**
Indicates how many objects at one end of an association may communicate with another object at the other end of the association.
<table>
<thead>
<tr>
<th><strong>Non-functional requirement</strong></th>
<th>A requirement of a system that states the quality of the system, such as scalability.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Normalization</strong></td>
<td>A process of optimizing relational databases to improve the management of data.</td>
</tr>
<tr>
<td><strong>Object</strong></td>
<td>The basic building block of object-oriented systems. An object is defined by its attributes, which record the state of the object, and its operations, which define the behaviour of the object.</td>
</tr>
<tr>
<td><strong>Object orientation</strong></td>
<td>An approach to the analysis, design and construction of software systems that is based on the notion of objects.</td>
</tr>
<tr>
<td><strong>OMG</strong></td>
<td>The Object Management Group, an organization that was formed to create a component-based software market place through the introduction of standardized object software. It has been responsible for the development of various standards such as UML and CORBA. See <a href="http://www.omg.org/">http://www.omg.org/</a>.</td>
</tr>
<tr>
<td><strong>ORB</strong></td>
<td>Object Request Broker – a software component that services requests and responses between objects in a distributed environment.</td>
</tr>
<tr>
<td><strong>Pattern</strong></td>
<td>A commonly used design that may be adapted to particular circumstances.</td>
</tr>
<tr>
<td><strong>Polymorphism</strong></td>
<td>The property of objects that two objects of different classes may use the same names for operations and attributes without creating ambiguity.</td>
</tr>
<tr>
<td><strong>Postcondition</strong></td>
<td>A statement of the legal results of an operation or use case.</td>
</tr>
<tr>
<td><strong>Precondition</strong></td>
<td>A statement of the legal conditions under which an operation or use case may fire.</td>
</tr>
<tr>
<td><strong>Primary path</strong></td>
<td>The most commonly used route through a business process or use case.</td>
</tr>
<tr>
<td><strong>Process map</strong></td>
<td>A high-level description of a business, grouping together the key processes and sub-processes of a business.</td>
</tr>
<tr>
<td><strong>Project manager</strong></td>
<td>An individual with overall responsibility for the project, whose role is to plan, resource, monitor deliverables and quality, manage risk and report to the steering group on progress.</td>
</tr>
<tr>
<td><strong>Prototype</strong></td>
<td>A quickly produced system used to illustrate a point, generate discussion or test an idea.</td>
</tr>
<tr>
<td><strong>RAD</strong></td>
<td>Rapid Application Development, a lightweight method that uses iterative development with frequent releases of versions of software.</td>
</tr>
<tr>
<td><strong>Relational database</strong></td>
<td>A database made up of a number of tables, where each table is constructed of records with simple fields, with a fixed number of column headings.</td>
</tr>
<tr>
<td><strong>Reliability</strong></td>
<td>A non-functional requirement indicating how well the system performs in terms of availability and prevention of damage to data.</td>
</tr>
<tr>
<td><strong>Requirements analysis</strong></td>
<td>The process of providing a definition of requirements of a system, usually expressed as a set of use cases.</td>
</tr>
</tbody>
</table>
Return on Capital  A financial measure to estimate the overall performance of a business defined by the equation
\[
\frac{\text{Revenue} - \text{Expenditure}}{\text{Capital Employed}} \times 100
\]

RFI  Request For Information. A formal document sent to a supplier asking for information about their product.

RFT  Request For Tender. A formal document sent to a supplier, asking them to tender for business.

Scalability  A non-functional requirement indicating the need for a system to deal with a number of users or transactions in a given period.

Scenario  A single execution of a business or of a system.

Second normal form  A table is in second normal form if every non-key attribute is fully dependent on the whole of the primary key, and not on part of the primary key.

Sequence diagram  A UML diagram that shows how the execution of a scenario in a system is implemented in terms of collaborations between objects.

Serializability constraint  The property that two database transactions must operate as if one completed before the other.

Stakeholder  Anyone who has any involvement in the initiation, specification, use, funding or management of a system.

Stakeholder analysis  A component of requirements that describes the stakeholders in the system, their roles and goals.

Steering group  A group responsible for the setting of project goals, control of funding and monitoring of achievement.

SWOT analysis  An analysis of the strengths, weaknesses, opportunities and threats of a situation or proposal.

Systems analysis  The process of providing a comprehensive definition of the external behaviour of a system, in terms of its interaction with the outside world and the types of information it needs to represent.

System test  A suite of tests to determine the acceptability of a system in terms of meeting its defined specification.

Technical architecture  The technical components of a system, such as the servers and networks, and major system software items such as databases.

Third normal form  A database table is in third normal form if it is in second normal form and there are no transitive dependencies.

Time box  A fixed period of time in which to carry out an activity. Usually the activity sets off with a prioritized list of actions or goals, and an agreed resource. Once the time period approaches expiry, the activity is brought to a conclusion.

Transitive dependency  An attribute A is said to be transitively dependent on B if A is dependent on C and C is dependent on B.
| **Unit test** | A test to determine the correct functioning of a part of a system, such as an individual use case. |
| **USDP** | The Unified Software Development Process, a staged process using a series of intermediate models in the production of software. |
| **Use case** | A well-defined, meaningful piece of functionality in a system. |
| **Use case description** | A detailed description of a use case in terms of its supplied functionality and any particular non-functional requirements it must conform to. |
| **Use case diagram** | A UML diagram showing relationships between actors and use cases. |
| **Use case realization** | A set of objects and their interactions designed to provide the functionality of a system. |
| **User acceptance test** | A suite of tests to determine the acceptability of a system in terms of meeting the expectations and needs of the users. |