Terms and Acronyms

This section defines terms and acronyms used in Construx's Requirements Boot Camp Seminar. Cross-references to other terms defined in this section are denoted by *italicized* text.

Term	Acronym	Definition / Description
Actor		An actor is a person, organization, or external system that plays a role in one or more interactions with your system (actors are typically drawn as stick figures on *Use Case* diagrams.
Actor Catalog		See *Actor Table.*
Actor Hierarchy		See *Actor Map.*
Actor Map		Defines the relationships among the *actors* in the *actor table* in terms of how their roles are shared and disparate. Also known as: *Actor Hierarchy*
Actor Table		Defines the roles played by people and things that will interact directly with the system. The contents of an actor table typically include names, brief descriptions, physical locations, and necessary job aids. Also know as: *Actor Catalog, Direct Users List*
Ambiguity		Doubtfulness or uncertainty of meaning or intention. The condition of admitting more than one meaning.
Artifact		The tangible result of work performed. May be used at any level of detail, e.g., the artifact resulting from a task might be a document, while the artifact resulting from a project might be a software system.
Atomic Business Rules		See *Business Rules.*
Business Policies		Define the principles and regulations that influence the behavior and structure of the business being modeled. Business policies provide the useful basis for decomposing *business rules.* Also know as: *Guidelines, High-Level Business Rules, Standards.*
Business Requirement		High-level objectives of the organization or customer requesting a system or product. Also known as the *Why Requirements*.

Term	Acronym	Definition / Description
Business Rules		A statement that defines or constrains some aspect of the business. Business rules must be a term, fact, constraint, or derivation. Also know as: *Free-form business rules, atomic business rules,* constraints, or invariants.
Casual Use Case		A *use case* written as a simple, prose paragraph. It is likely to be missing some project information associated with the use case and it is likely to be less rigorous in its description than the *fully dressed* version.
Conceptual Class Model		A *domain model* that shows the relationships of logical business objects in the system.
Conceptual Data Model		A *domain model* that shows the relationships of logical data entities in the system. Also known as *Logical Entity Relationship Diagram*.
Construction Lead		Plans and directs the construction, integration, product builds, development environment, and deployment activities. See *Team of Leads.*
Context Diagram		Shows the system as a whole in its environment. In a context diagram, viewers see the system as a black box that manipulates inputs and produces output in a way that's not visible or controllable. Also known as: *Context-level use case, level-zero data flow diagram*, context-level use case, system-level use case.
Context-Level Use Case		See *Context Diagram.*
Decision Table		Describes the conditions associated with particular actions or decisions, along with the constraints on the associated behavior. A decision table shows rules as rows and conditions as columns, with an "X" in each cell for each action that should be taken by the person performing the behavior. Also known as: *Decision Tree.*
Decision Tree		Another way of formatting a *decision table* using flow chart type notation.
Design Lead		Plans and directs the system architecture and design activities. See *Team of Leads.*

Term	Acronym	Definition / Description
Dialog Map		Shows the layout of the user interface (screens, windows, dialog boxes, HTML pages) and the possible navigation paths among the interface elements.
Direct Users List		See *Actor Table.*
Domain Model		Defines groups of information that must be stored in the system and the relationships among the groups. The model contains at least a list of business objects, data entities, or classes. It can also diagram the relationships among them. The detailed domain model should also contain attributes for each item. Also known as: *Conceptual Class Model, Conceptual Data Model, Logical Entity Relationship Diagram.*
Event List		See *Event Table.*
Event Table		Defines the triggers of events to which a system responds. Some of these responses involve visible outputs; others change only internal information; others do both. Also known as: *Event List.*
Free-Form Business Rules		See *Business Rules.*
Fully Dressed Use Case		A fully specified *use case*. Includes the pre and post conditions and a list of steps that shows the sequence of dialogs or interactions between the *actor* and the system that leads from the preconditions to the post conditions.
Functional Requirement		Functionality that must be built into the system to satisfy the business requirements. Also known as a *what requirement.*
Gist		A rough, informal, brief description or summary. See: Tom Gilb, *Competitive Engineering: A Handbook for Systems and Software Engineering Management Using Planguage.*
Glossary		Defines the meanings of all business terms relevant to the system being built. These serve as the foundation for all requirements models and business rules; the goal of the glossary is to provide a common vocabulary on which all of the stakeholders agree. Also known as: Terms and Acronyms.
Goals		The desired results of a *project*, *process*, or *activity*. In general, software projects exist to meet business or organizational goals through the creation of software. Goal and *objective* are synonyms in CxOne.

Term	Acronym	Definition / Description
High-Level Business Rules		See *Business Policies.*
How Requirement		Often a *non-functional requirement*. Sometimes used to refer to design issues.
How Well Requirement		See *non-functional requirement*.
Level-Zero Data Flow Diagram		See *Context Diagram.*
Logical Entity Relationship Diagram		See *Conceptual Data Model.*
Low-Level Business Rules		See *Business Rules.*
Non-functional Requirement		Captures requirements such as compatibility, usability, performance, reliability, etc. Also known as the *how* or *how well requirements.*
Objectives		See *goals*.
Planning and Tracking Lead		Directs overall flow of technical work on the project. Directly responsible for project planning and overseeing the execution of work breakdown, estimation, scheduling, and tracking. See *Team of Leads.*
Process Map		Shows the organizational context in terms of the flows of inputs and outputs across functions or roles for a specific work process. Also known as: *Swim lane diagram, workflow map.*
Progressive Elaboration		Developing thoroughly in steps, and continuing steadily by increments. For example, the project scope will be broadly described early in the project, and made more explicit and detailed as the project team develops a better and more complete understanding of the objectives and deliverables. Progressive elaboration should not be confused with scope creep.

Term	Acronym	Definition / Description
Project Business Manager		Responsible for successful business outcome of the project. In charge of project staffing, acquiring resources for the project, personnel issues, top-level work assignments, and stakeholder interaction. Is the top decision maker on the project, but normally defers technical decisions to the appropriate technical lead. Resolves disputes between project participants. Responsible for coordinating the activities with the stakeholders, project sponsor, and project reviewers. See *Team of Leads.*
Prototype		Anything that captures the look and feel of the user interface to be built for the new system and allows users to play with it. A prototype is created to explore the user interface functionality or look and feel.
Quality Lead		Plans and directs all quality assurance and quality control activities including reviews and testing. See *Team of Leads.*
Relationship Map		Shows the organizational context in terms of the relationships that exist among the suppliers and external customers. In particular, the diagram shows the functions that outsiders and your organization perform and the inputs and outputs from these functions. Also known as: Organizational context diagram.
Requirement		A detailed description of what the software is supposed to do.
Requirements		Software requirements. Determining and capturing what a software system should do.
Requirements Lead		Plans and directs activities for discovering, documenting and maintaining the set of requirements. See *Team of Leads.*
Scenario		A formal description of the flow of events that occurs during a Use Case instance. This can include stories, examples, and drawings of GUIs.
Scope		The boundaries of the project.

Term	Acronym	Definition / Description
Simplified English		Simplified English uses a limited vocabulary and a set of writing rules. The writer can only use words on the approved list and technical terms (words common to the field). Each word on the list has only one approved definition. Note: Simplified English was originally developed for Aerospace manufacturers to facilitate the use of maintenance manuals by non-native speakers of English. However, it can be applied to any industry.
Scope Model		See *Context Diagram.*
Software Product Line		A set of software-intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way.
Stakeholder		A person, group, or organization that is actively involved in a project, is affected by its outcome, or can influence its outcome.
Stakeholder Class		A set of stakeholders that share a common characteristics. • May be human, machine, or both • May be directly or indirectly impacted
State Chart Diagram		See *State Diagram.*
State Diagram		Define how time affects your *domain model* in terms of the possible states that elements of the model can assume and the transitions between those states.
State Model		See *State Diagram.*
Swim lane diagram		See *Process Map.*
System-Level Use Case		See *Context Diagram.*
Team of Leads		A small set of leads working in interdependent multidisciplinary roles. The responsibilities for planning and managing the work are split between the technical leads and the project business manager. See *Project Business Manager, Planning and Tracking Lead, Requirements Lead, Quality Lead, Design Lead,* and *Construction Lead.*
Use Case		Describe the major functions that the system will perform for external *actors*, and also the goals that the system achieves for those *actors* along the way. A single use of the system depicted as an interaction between the user and the system.

Term	Acronym	Definition / Description
Use Case Brief		A brief synopsis of a *use case.*
Use Case Group		See *Use Case Package.*
Use Case Map		Illustrates the predecessor and successor relationships among *use cases*.
Use Case Package		Cohesive groups of *use cases* that can form things such as a logical architecture, a test group, or a release of the system.
Use Case Specification		The documentation required to capture a use case.
User Class		See *Stakeholder Class.*
User Interface	UI	The interface between a system and its external users. Normally used to describe interaction of the system with human operators.
User Interface Navigation Diagram		See *Dialog Map.*
User Interface Prototype		See *prototype*. A prototype created to explore the user interface functionality or look and feel.
User Interface Storyboard		See *Dialog Map.*
Walkthrough		An *informal review* in which the *author* and one or more people meet to review an *artifact* with the intent of finding *defects*.
What Requirement		See *Functional Requirement.*
Why Requirement		See *Business Requirement.*
Work Product		See *artifact*.
Workflow map		See *Process Map.*