**FORUMS**

1. **Business Analysis** – It is the practice of enabling change in an enterprise by defining needs and recommending solutions that deliver value to stakeholders.
2. **Requirement** – A Requirement is the need of the client. This Need or Requirement will transform into a Solution while taking various shapes and forms as it progresses from each stage of SDLC.
3. **Stakeholder** – A ‘Stakeholder’ is any person or group of persons or an organization that is directly or indirectly effected or impacted by the proposed IT Solution.
4. **Business Process Model -** A **Business Process Model (BPM)** is a visual representation or diagram that outlines the steps and activities involved in a business process. It helps organizations understand, analyze, and improve their workflows, ensuring they run efficiently and effectively.
5. **Scope Creep –** If any project cannot be completed within the Budget and Time constraints, we say that Scope creeps.
6. **Scrum** – Scrum is an iterative development methodology used to manage software projects. In scrum-based projects, there isn’t a specific project manager directing project team tasks, the team is self-directed with co-located team members relying on communication over documentation for effective project delivery.
7. **Sprint** – A sprint is a Scrum-based agile methodology concept that is similar to an iteration. A sprint is time boxed to deliver a specific set of user stories and produce working features within a set time period.
8. **Story Points** – A story point is a relative estimation method used to determine the size of user stories so teams can determine how much work can be done during an iteration.
9. **Product Owner** – PO is responsible for product vision, requirement questions, considers stakeholder interest. He decides whether to accept or reject product increment, continue or discontinue the development and whether to ship the product or not.
10. **Scrum Master** – He will facilitate the scrum process by resolving issues, creating an environment for team self-organization and capturing empirical data to adjust forecast. The Scrum Master will not have any management authority over the team.
11. **Use Case Diagram** – This is a high-level diagram and mother of all diagrams. The main focus of this diagram will be on ‘how external interfaces’ will be interacting with the proposed IT system.  
    Use case diagrams can be used to describe the functionality of a system horizontally.
12. **Actor** – Actor is a living or non-living thing. Actor is represented by a noun. An actor always stays away from the system boundary.  
    Primary actor initiates the system to work and the system depends on the secondary actor for information.
13. **Domain Model -** A Domain Model is a conceptual representation that defines the structure, relationships, and behaviours of entities within a specific problem domain.

It is also known as conceptual modelling. It depicts the concepts(idea, thing, or object) that are easily identifiable in the problem description.

The domain model primarily serves by transferring information between the external commerce system.

1. **Sequence diagram -** Sequence Diagram is a type of interaction diagram used in software engineering and system design to illustrate how processes operate with one another and in what order. A sequence diagram can map scenarios described by a use case in step-by-step details to define how classes collaborate to achieve your application’s goal.  
   Time is represented in a Vertical direction showing the sequence of interactions of the header elements, which are displayed horizontally at the top of the diagram.
2. **MVC architecture -** The Model-View-Controller (MVC) framework is an architectural pattern that separates an application into three main logical components Model, View and Controller.  
   View represents the presentation layer of the application.

The model represents the data and the business logic of the application.

The controller acts as an intermediary between the Model and the View.

To identify classes from the use case diagram, we apply MVC rules on each use case to derive classes.

1. **Conflict management -** Conflict Management refers to the process of handling, resolving or mitigating disputes and disagreements in a way that promotes cooperation, understanding and productive outcomes.

Conflict management is the use of techniques to resolve disagreements or control the level of discord.

Conflict resolution techniques include facilitating meetings for the conflicting parties to identify the problem, discuss resolutions and create superordinate goals that require cooperation from conflicting parties.

1. **Data Mapping -** Data mapping is the process of connecting data from one source to another. It is like creating a guide or map that shows how data in one place corresponds to data in another place.

Data mapping involves mapping data from its source to the related objective fields. For instance, mapping a name of an individual from one data source to their email, phone or address fields in another record.

1. **Agile Manifesto -** The Agile Manifesto was created in 2001 by a group of software developers and practitioners who wanted to improve the way the software was developed.

The Agile Manifesto outlines the values and principles that we must consider while accepting Agile. It emphasizes flexibility, collaboration and customer-centricity.

It has four main values and twelve principles.

1. **User Stories** - User Story is a very small task that can be delivered in a small timeframe. It is a simple, informal description of a feature from the perspective of the end user. User stories help to prioritize work, guide development and serve as a communication tool between stakeholders. The activities that developers need to complete the user story are called Tasks. The value statement is the description of the user story.
2. **Business Value -** Business Value is how important is this feature (user Story) to the Business. This is not the cost of Development or the complexity of the feature. This is estimated by Scrum Currency Notes.
3. **Complexity Points -** CP is also known as Story Points (SP). CP is the effort required by the Scrum Developers to develop this feature (user story) using technology. Efforts include time taken to solve the complexity and write the code. CP is estimated by the Scrum Developers by using Poker cards.
4. **Epic** - An Epic is a set of related user stories. They are also considered a ‘really big user story’. It represents a significant feature or functionality. Epics are high-level, often spanning multiple sprints or iterations, and they provide a way to organize and prioritize work in a product backlog.
5. **Impediment Log** - An Impediment Log is also known as an issue log or obstacle log. It is a document or tool used in Agile Software Development to track and manage obstacles, bottlenecks or any factors that impede the progress of the project or team.