

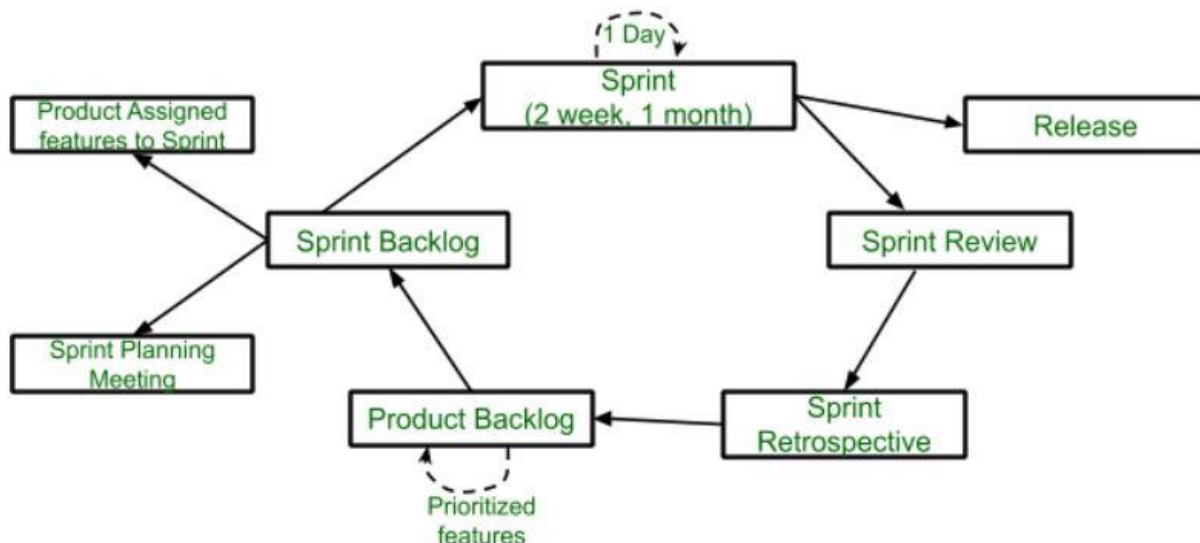
## Scrum-Crystal-Feature Driven Development (FDD)

### Scrum:

Scrum is the type of Agile framework. It is a framework within which people can address complex adaptive problem while productivity and creativity of delivering product is at highest possible values. Scrum uses Iterative process. Silent features of Scrum are:

- Scrum is light-weighted framework
- Scrum emphasizes self-organization
- Scrum is simple to understand
- Scrum framework help the team to work together

### Lifecycle of Scrum:



**Sprint:** A Sprint is a time box of one month or less. A new Sprint starts immediately after the completion of the previous Sprint.

**Release:** When the product is completed, it goes to the Release stage.

**Sprint Review:** If the product still has some non-achievable features, it will be checked in this stage and then passed to the Sprint Retrospective stage.

**Sprint Retrospective:** In this stage quality or status of the product is checked.

**Product Backlog:** According to the prioritize features the product is organized.

**Sprint Backlog:** Sprint Backlog is divided into two parts Product assigned features to sprint and Sprint planning meeting.

## **Crystal:**

Crystal methods in Agile Development/Framework: The crystal method is an agile framework that is considered a lightweight or agile methodology that focuses on individuals and their interactions. The methods are color-coded to significant risk to human life. It is mainly for short-term projects by a team of developers working out of a single workspace. Among a few Agile Software Development Life Cycle (SDLC) model's crystal is considered as one of the Agile SDLC models.

### **Properties of Crystal Agile Framework:**

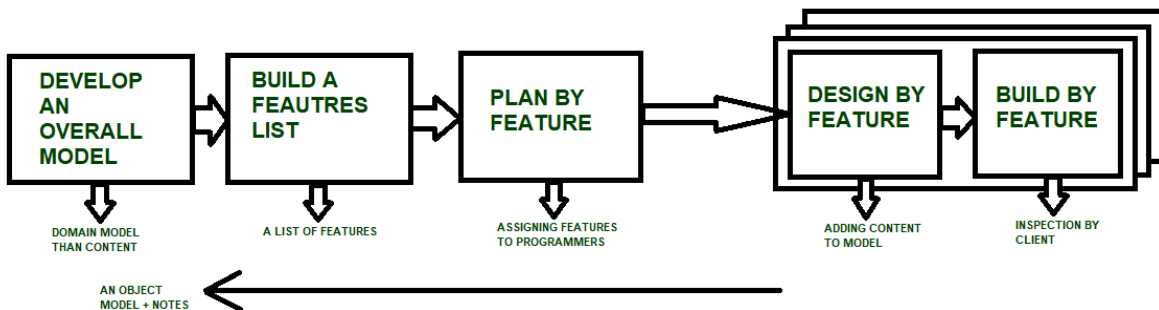
1. Frequent Delivery- It allows you regularly deliver the products and test code to real users. Without this, you might build a product that nobody needs.
2. Reflective Improvement- No matter how good you have done or how bad you have done. Since there are always areas where the product can be improved, so the teams can implement to improve their future practices.
3. Osmotic Communication- Alistair stated that having the teams in the same physical phase is very much important as it allows information to flow in between members of a team as in osmosis.
4. Personal Safety- There are no bad suggestions in a crystal team, team members should feel safe to discuss ideas openly without any fear.
5. Focus- Each member of the team knows exactly what to do, which enables them to focus their attention. This boosts team interaction and works towards the same goal.
6. Easy access to expert users- It enhances team communication with users and gets regular feedback from real users.
7. Technical tooling- It contains very specific technical tools which to be used by the software development team during testing, management, and configuration. These tools make it enable the team to identify any error within less time.
8. Continuous learning – The framework emphasizes on continuous learning, enabling team members to acquire new skills and knowledge, and apply them in their work.
9. Teamwork – The framework stresses on the importance of teamwork, promoting collaboration, and mutual support among team members.
10. Timeboxing – The framework adopts timeboxing to manage project deadlines, ensuring that the team delivers within set timelines.
11. Incremental development – The framework promotes incremental development, enabling the team to deliver working software frequently, and adapt to changes as they arise.
12. Automated testing – The framework emphasizes on automated testing, enabling the team to detect and fix bugs early, reducing the cost of fixing errors at later stages.
13. Customer involvement – The framework emphasizes on involving customers in the development process, promoting customer satisfaction, and delivering products that meet their needs.
14. Leadership – The framework encourages leadership, enabling team members to take ownership of their work and make decisions that contribute to the success of the project.

## Feature Driven Development(FDD)

FDD stands for Feature-Driven Development. It is an agile iterative and incremental model that focuses on progressing the features of the developing software. The main motive of feature-driven development is to provide timely updated and working software to the client.

### FDD Lifecycle

- Build overall model
- Build feature list
- Plan by feature
- Design by feature
- Build by feature



**Develop a model-** The overall model is created by the chief architect, or another professional leading the project, by identifying the scope and context of the system.

**Build a list of features-** Developers brainstorm a list of potential items that would be useful to users and could be completed along a set timeline for release. Each feature should be manageable within a timeframe of around two weeks.

**Plan out each feature-** Features are organized by how long they take to create and how important they are to the client. Ownership of each feature is also assigned.

**Design each feature-** The actual details of each feature are produced, inspected and finalized.

**Build each feature-** After the design is improved, the completed feature is added to the official build for delivery to the client.

## **Characteristics of FDD**

- Short iterative: FDD lifecycle works in simple and short iterations to efficiently finish the work on time and gives good pace for large projects.
- Customer focused: This agile practice is totally based on inspection of each feature by client and then pushed to main build code.
- Structured and feature focused: Initial activities in lifecycle builds the domain model and features list in the beginning of timeline and more than 70% of efforts are given to last 2 activities.
- Frequent releases: Feature-driven development provides continuous releases of features in the software and retaining continuous success of the project.