**Crystal methods in Agile Development/Framework**

**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)](https://www.geeksforgeeks.org/software-development-life-cycle-sdlc/) models crystal is considered as one of the Agile SDLC models.
Two core beliefs of the Crystal method:

* Find your own way and methods to optimize workflow.
* Make use of unique methods to make the project unique and dynamic.

**Let’s know about the history of the Crystal Method**: The crystal method was developed by an American scientist named Alistair Cockburn who worked at IBM. He decided not to focus on step-by-step developmental strategies, but to develop team collaboration and communication. Some of the traits of Cockburn’s Crystal method were:

* Human-powered i.e. the project should be flexible and people involved in preferred work.
* Adaptive i.e. approaches don’t have any fixed tools but can be changed anytime to meet the team’s specific needs.
* Ultra-light i.e. this methodology doesn’t require much documentation.

**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.

**How does Crystal function?**

Till now, we got to know that crystal is a family of various developmental approaches, and it is not a group of prescribed developmental tools and methods. In the beginning, the approach is set by considering the business requirements and the needs of the project. Various methodologies in the Crystal family also known as weights of the Crystal approach are represented by different colors of the spectrum.
Crystal family consists of many variants like Crystal Clear, Crystal Yellow, Crystal Red, Crystal Sapphire, Crystal Red, Crystal Orange Web, and Crystal Diamond.

1. **Crystal Clear-** The team consists of only 1-6 members that is suitable for short-term projects where members work out in a single workspace.
2. **Crystal Yellow-** It has a small team size of 7-20 members, where feedback is taken from Real Users. This variant involves automated testing which resolves bugs faster and reduces the use of too much documentation.
3. **Crystal Orange-** It has a team size of 21-40 members, where the team is split according to their functional skills. Here the project generally lasts for 1-2 years and the release is required every 3 to 4 months.
4. **Crystal Orange Web-** It has also a team size of 21-40 members were the projects that have a continually evolving code base that is being used by the public. It is also similar to Crystal Orange but here they do not deal with a single project but a series of initiatives that required programming.
5. **Crystal Red-** The software development is led by 40-80 members where the teams can be formed and divided according to requirements.
6. **Crystal Maroon-** It involves large-sized projects where the team size is 80-200 members and where methods are different and as per the requirement of the software.
7. **Crystal Diamond & Sapphire-** This variant is used in large projects where there is a potential risk to human life.

The below figure illustrates about crystal team



CRYSTAL FAMILY (TEAM MEMBERS)

**Benefits of using the Crystal Agile Framework :**

* Facilitate and enhance team communication and accountability.
* The adaptive approach lets the team respond well to the demanding requirements.
* Allows team to work with the one they see as the most effective.
* Teams talk directly with each other, which reduces management overhead.
* **Faster delivery –** The framework enables the team to deliver working software faster, which can help gain a competitive advantage in the market.
* **Higher quality –** The framework emphasizes on quality, enabling the team to detect and fix defects early in the development process, resulting in a higher quality product.
* **Improved customer satisfaction –** The framework promotes customer involvement, enabling the team to deliver products that meet customer needs, resulting in higher customer satisfaction.
* **Increased productivity –** The framework enables the team to focus on delivering the highest value features, which can increase productivity and reduce waste.
* **Flexibility –** The framework is highly adaptable, enabling the team to adjust to changing requirements, and make decisions based on real-time feedback.
* **Empowerment –** The framework promotes empowerment, enabling team members to take ownership of their work, and make decisions that contribute to the success of the project.
* **Reduced risk –** The framework promotes risk management, enabling the team to identify and mitigate potential risks early in the development process, reducing the likelihood of project failure.

**Drawbacks of using the Crystal Agile Framework :**

* A lack of pre-defined plans may lead to confusion and loss of focus.
* Lack of structure may slow down inexperienced teams.
* Not clear on how a remote team can share knowledge informally.
* **Lack of predictability –** The framework’s emphasis on adaptability and flexibility may result in a lack of predictability, making it difficult to plan and estimate project timelines and budgets.
* **Lack of documentation –** The framework’s emphasis on communication and collaboration may result in a lack of documentation, making it difficult to track progress and maintain a record of decisions.
* **Limited scalability –** The framework may not be suitable for large or complex projects, as the lack of structure and predefined plans may make it difficult to manage teams at scale.
* **Dependence on team expertise –** The framework relies heavily on the expertise and skills of the development team, which may not be suitable for teams with limited experience or knowledge.
* **Lack of clarity on roles and responsibilities –** The framework’s emphasis on self-organizing teams may result in a lack of clarity on roles and responsibilities, leading to confusion and a loss of focus.
* **Inability to handle regulatory requirements –** The framework may not be suitable for projects with strict regulatory requirements, as the lack of documentation and structure may not meet compliance standards.
* **Potential for informal knowledge sharing –** The framework’s emphasis on osmotic communication may result in informal knowledge sharing, which may be difficult to track and monitor for accuracy and completeness.