Presentation on Extreme Programming

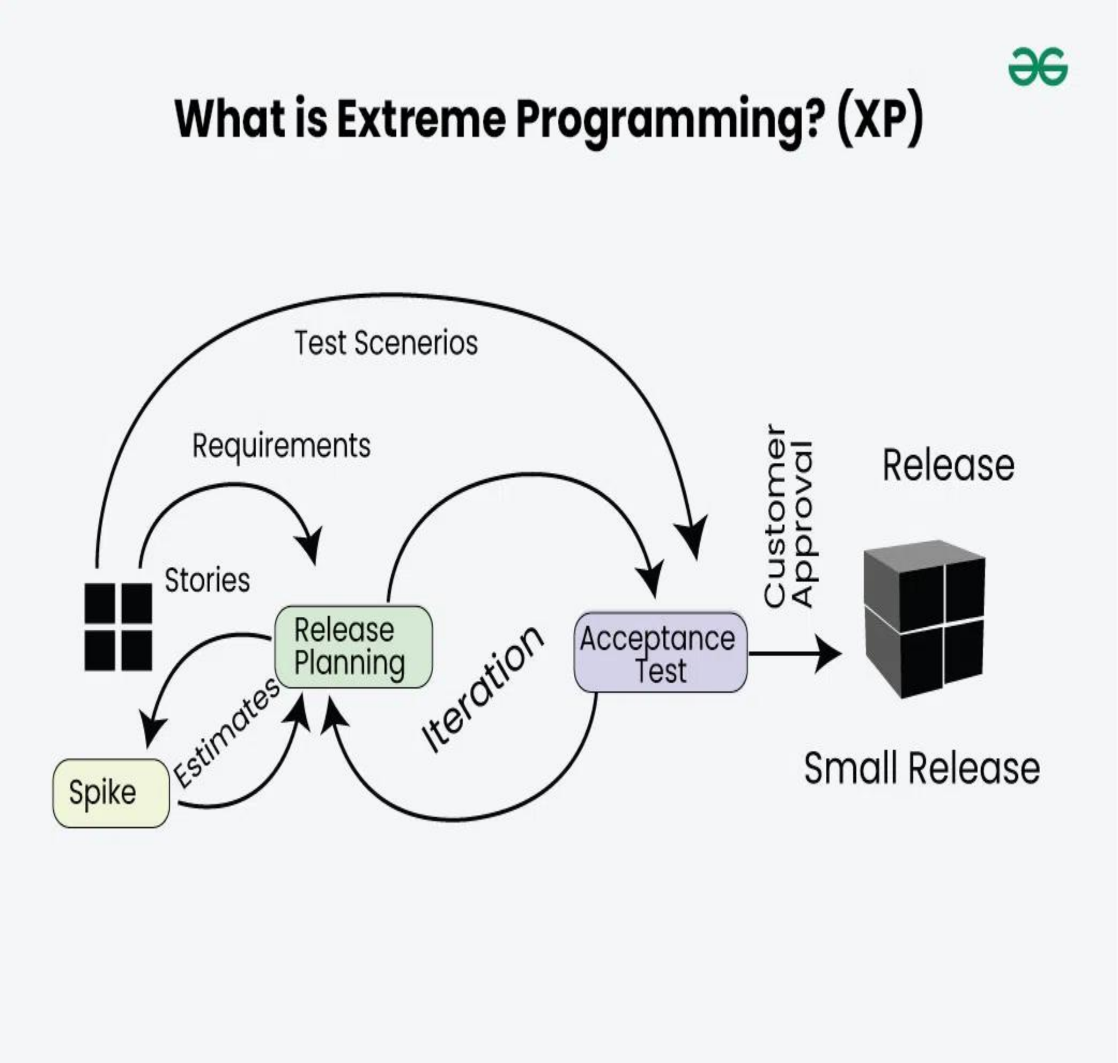
Course : Software Engineering

Submitted by
Name:Umma kulsum Ashamoni
ID: IT-21050
3rd Year 2nd semester

Submitted to
Dr. Ziaur Rahman
Assistant Professor
Dept of ICT

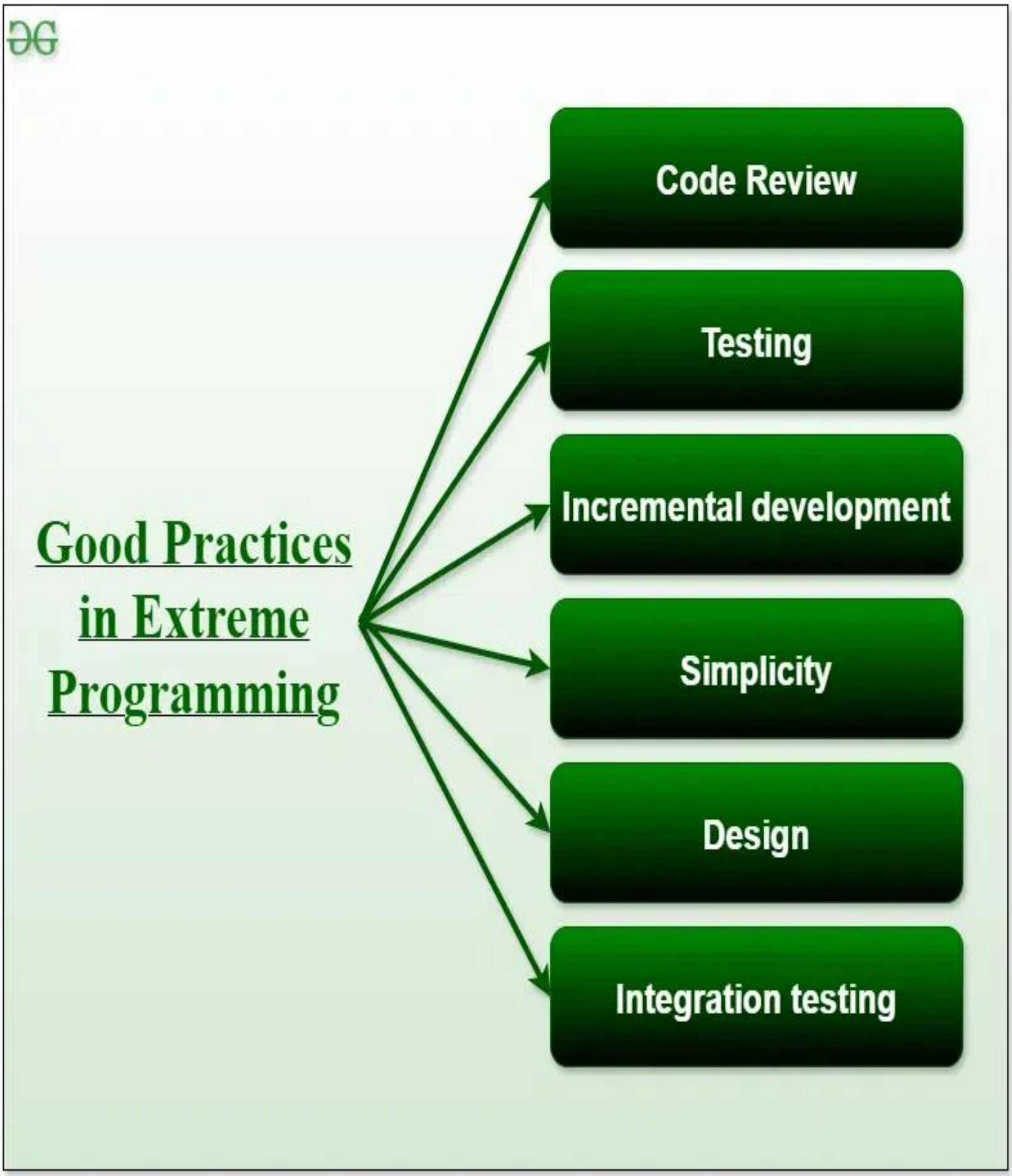
What is Extreme Programming (XP)?

Extreme Programming (XP) is an Agile software development methodology that focuses on delivering high-quality software through frequent and continuous feedback, collaboration, and adaptation. XP emphasizes a close working relationship between the development team, the customer, and stakeholders, with an emphasis on rapid, iterative development and deployment.



Good Practices in Extreme Programming

Some of the good practices that have been recognized in the extreme programming model and suggested to maximize their use are given below:



Good Practices in Extreme Programming

Code Review

Testing

Incremental development

Simplicity

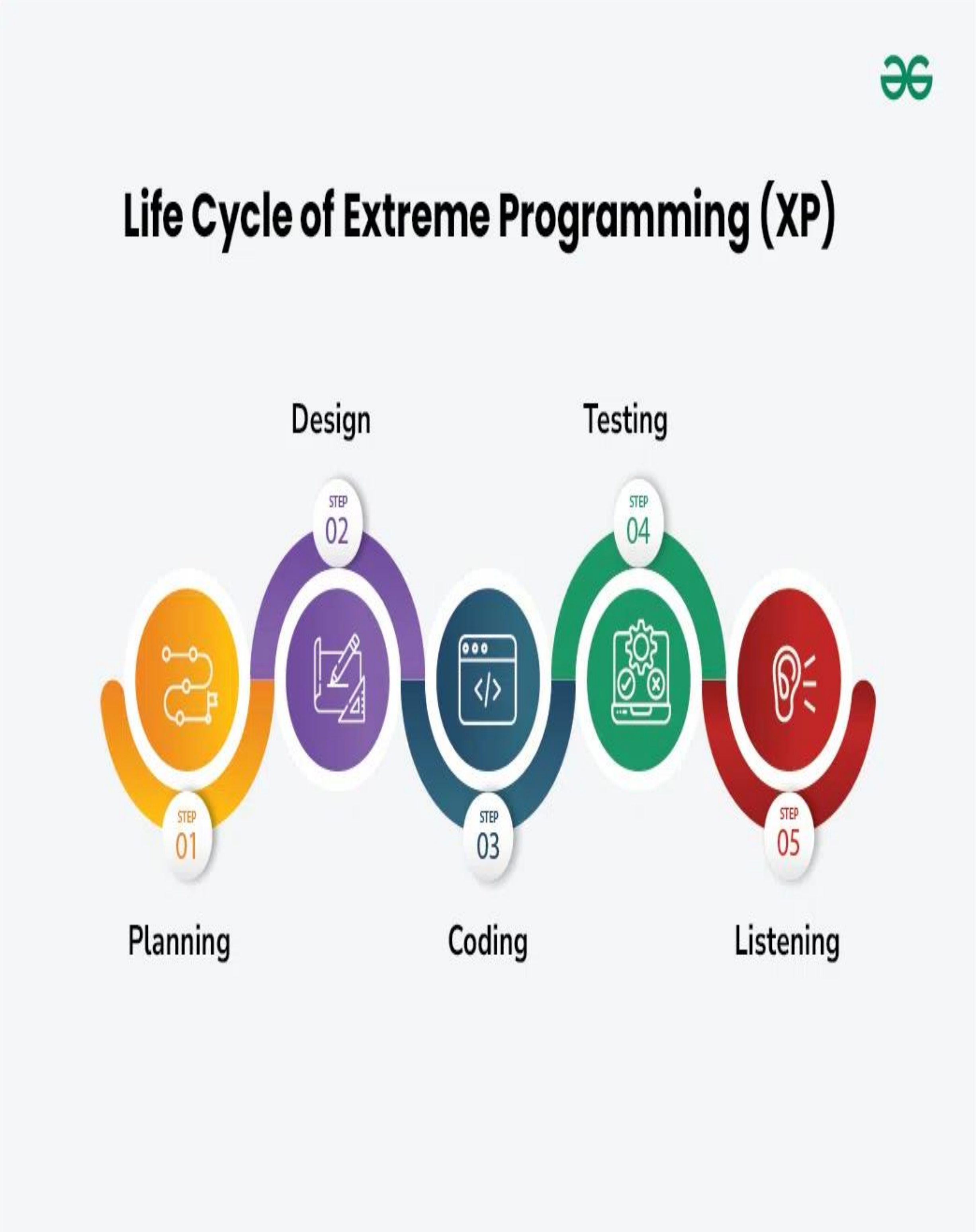
Design

Integration testing

- **ode Review:** Code review detects and corrects errors efficiently. It suggests pair programming as coding and reviewing of written code carried out by a pair of programmers who switch their work between them every hour.
- **Testing:** Testing code helps to remove errors and improves its reliability. XP suggests test-driven development (TDD) to continually write and execute test cases. In the TDD approach, test cases are written even before any code is written.
- **Incremental development:** Incremental development is very good because customer feedback is gained and based on this development team comes up with new increments every few days after each iteration.
- **Simplicity:** Simplicity makes it easier to develop good-quality code as well as to test and debug it.
- **Design:** Good quality design is important to develop good quality software. So, everybody should design daily.
- **Integration testing:** Integration Testing helps to identify bugs at the interfaces of different functionalities. Extreme programming suggests that the developers should achieve continuous integration by building and performing integration testing several times a day.

Life Cycle of Extreme Programming (XP)

The Extreme Programming Life Cycle consist of five phases:

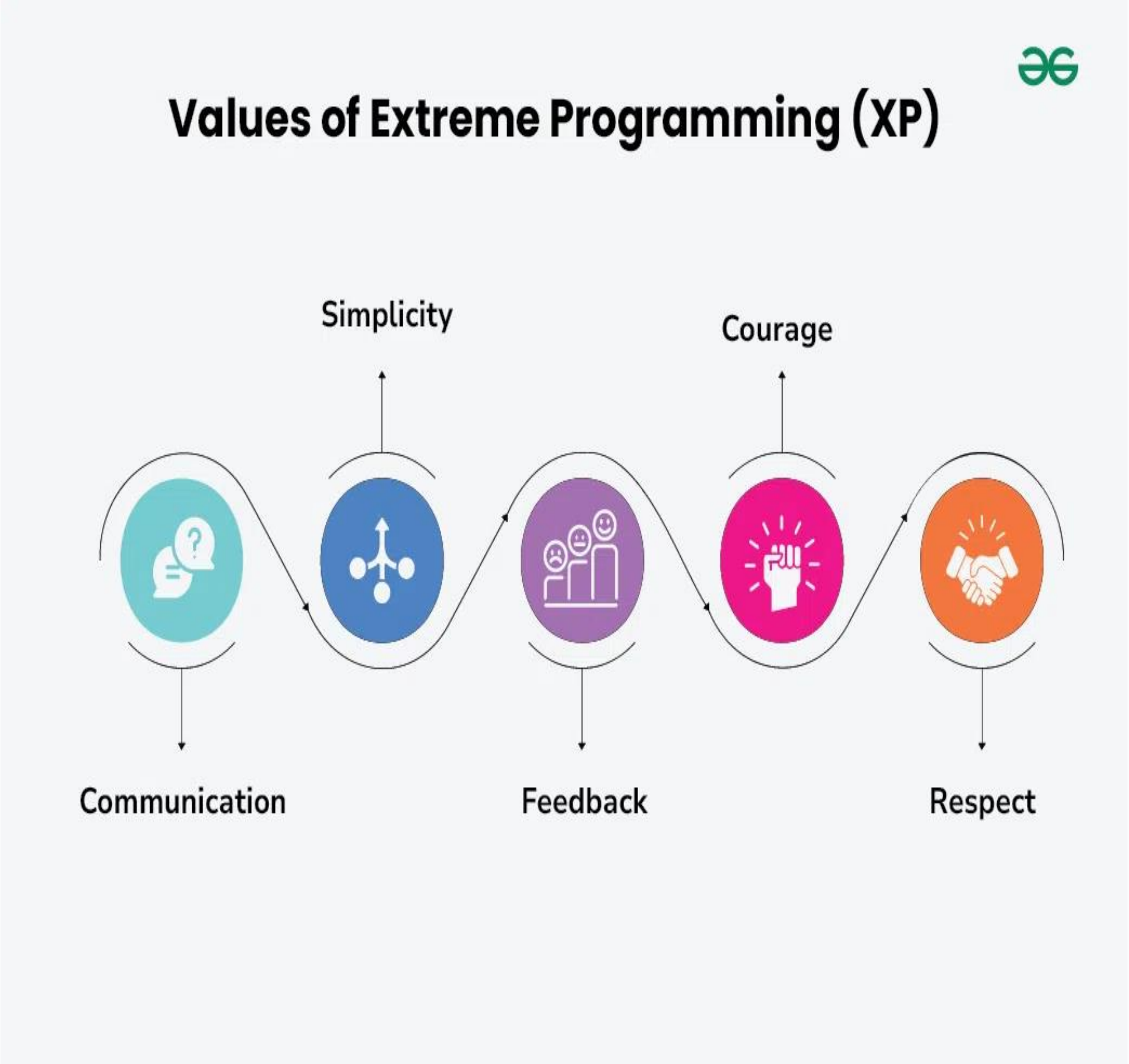


1. **Planning:** The first stage of Extreme Programming is planning. During this phase, clients define their needs in concise descriptions known as user stories. The team calculates the effort required for each story and schedules releases according to priority and effort.
2. **Design:** The team creates only the essential design needed for current user stories, using a common analogy or story to help everyone understand the overall system architecture and keep the design straightforward and clear.
3. **Coding:** Extreme Programming (XP) promotes pair programming i.e**.** wo developers work together at one workstation, enhancing code quality and knowledge sharing. They write tests before coding to ensure functionality from the start (TDD), and frequently integrate their code into a shared repository with automated tests to catch issues early.
4. **Testing:** Extreme Programming (XP) gives more importance to testing that consist of both unit tests and acceptance test**.** Unit tests, which are automated, check if specific features work correctly. Acceptance tests, conducted by customers, ensure that the overall system meets initial requirements. This continuous testing ensures the software's quality and alignment with customer needs.

5. **Listening:** In the listening phase regular feedback from customers to ensure the product meets their needs and to adapt to any changes.

Values of Extreme Programming (XP)

There are five core values of Extreme Program



1. **Communication:** The essence of communication is for information and ideas to be exchanged amongst development team members so that everyone has an understanding of the system requirements and goals. Extreme Programming (XP) supports this by allowing open and frequent communication between members of a team.

2. **Simplicity:** Keeping things as simple as possible helps reduce complexity and makes it easier to understand and maintain the code.

3. **Feedback:** Feedback loops which are constant are among testing as well as customer involvements which helps in detecting problems earlier during development.

4. **Courage:** Team members are encouraged to take risks, speak up about problems, and adapt to change without fear of repercussions.

5. **Respect**: Every member’s input or opinion is appreciated which promotes a collective way of working among people who are supportive within a certain group.

Advantages of Extreme Programming (XP)

- **Slipped schedules:** Timely delivery is ensured through slipping timetables and doable development cycles.
- **Misunderstanding the business and/or domain –** Constant contact and explanations are ensured by including the client on the team.
- **Canceled projects:** Focusing on ongoing customer engagement guarantees open communication with the consumer and prompt problem-solving.
- **Staff turnover:** Teamwork that is focused on cooperation provides excitement and goodwill. Team spirit is fostered by multidisciplinary cohesion.
- **Costs incurred in changes:** Extensive and continuing testing ensures that the modifications do not impair the functioning of the system. A functioning system always guarantees that there is enough time to accommodate changes without impairing ongoing operations.
- **Business changes:** Changes are accepted at any moment since they are seen to be inevitable.
- **Production and post-delivery defects:** the unit tests to find and repair bugs as soon as possible.

Applications of Extreme Programming (XP)

Some of the projects that are suitable to develop using the XP model are given below:

- **Small projects:** The XP model is very useful in small projects consisting of small teams as face-to-face meeting is easier to achieve.
- **Projects involving new technology or Research projects:** This type of project faces changing requirements rapidly and technical problems. So XP model is used to complete this type of project.
- **Web development projects:** The XP model is well-suited for web development projects as the development process is iterative and requires frequent testing to ensure the system meets the requirements.
- **Collaborative projects:** The XP model is useful for collaborative projects that require close collaboration between the development team and the customer.
- **Projects with tight deadlines:** The XP model can be used in projects that have a tight deadline, as it emphasizes simplicity and iterative development.
- **Projects with rapidly changing requirements: The** XP model is designed to handle rapidly changing requirements, making it suitable for projects where requirements may change frequently.
- **Projects where quality is a high priority:** The XP model places a strong emphasis on testing and quality

assurance, making it a suitable approach for projects where quality is a high priority.

XP, and other agile methods, are suitable for situations where the volume and space of requirements change are high and where requirement risks are consider*Values of Extreme Programming (XP)*

Conclusion

Extreme Programming (XP) is a Software Development Methodology, known for its flexibility, collaboration and rapid feedback using techniques like continuous testing, frequent releases, and pair programming, in which two programmers collaborate on the same code. XP supports user involvement throughout the development process while prioritizing simplicity and communication. Overall, XP aims to deliver high-quality software quickly and adapt to changing requirements effectively.