



Presentation On

Extreme Programming

Presented by

Tafhimul Afifa(IT-21035)



Mawlana Bhashani Science and Technology University

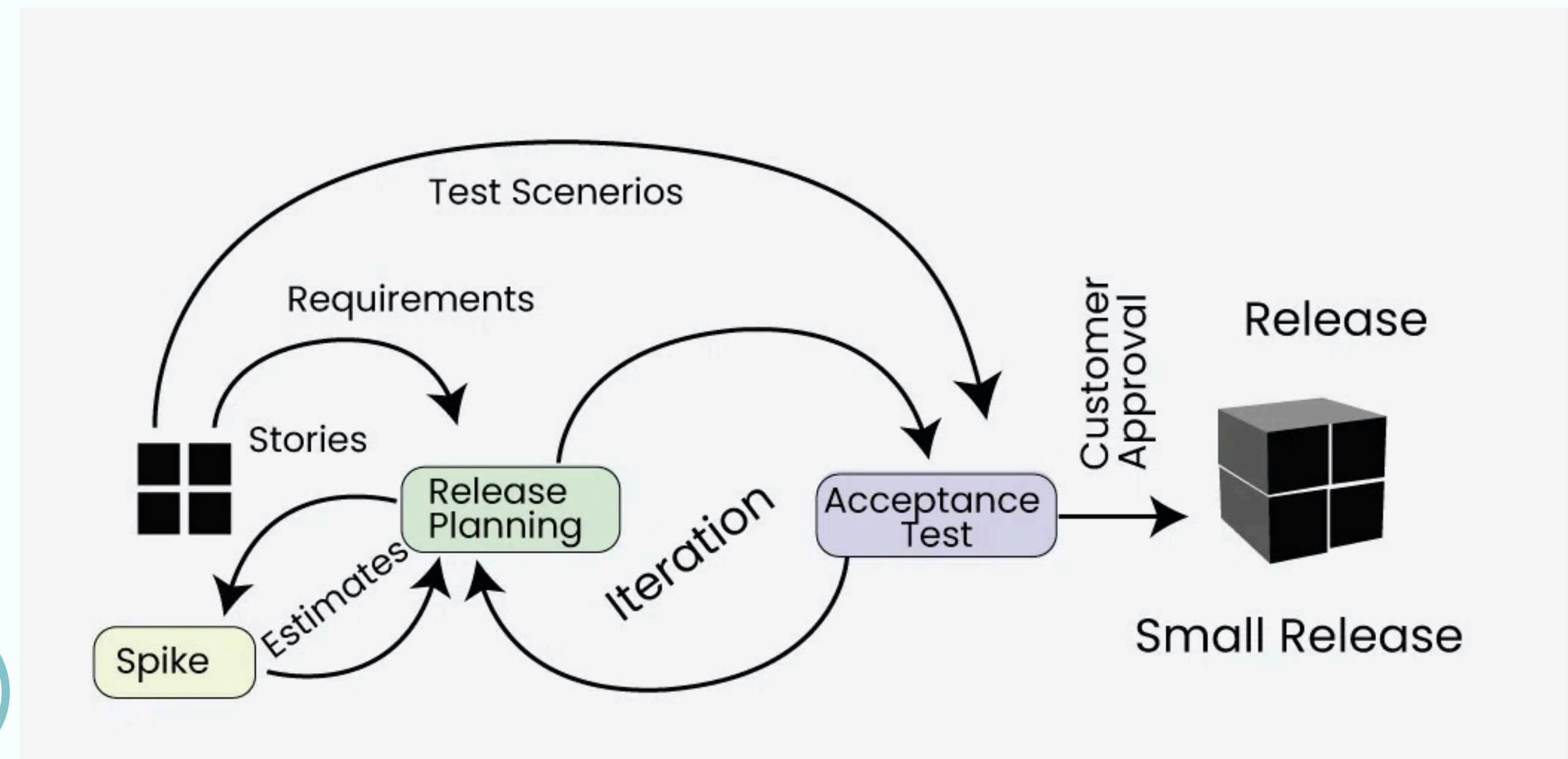
Course :Software Engineering

Course Code:ICT-3209

Date of Presentation:November 17, 2024

INTRODUCTION

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.



Core Values of XP

1. Communication
2. Simplicity
3. Feedback
4. Courage
5. Respect

Five Core Values of XP

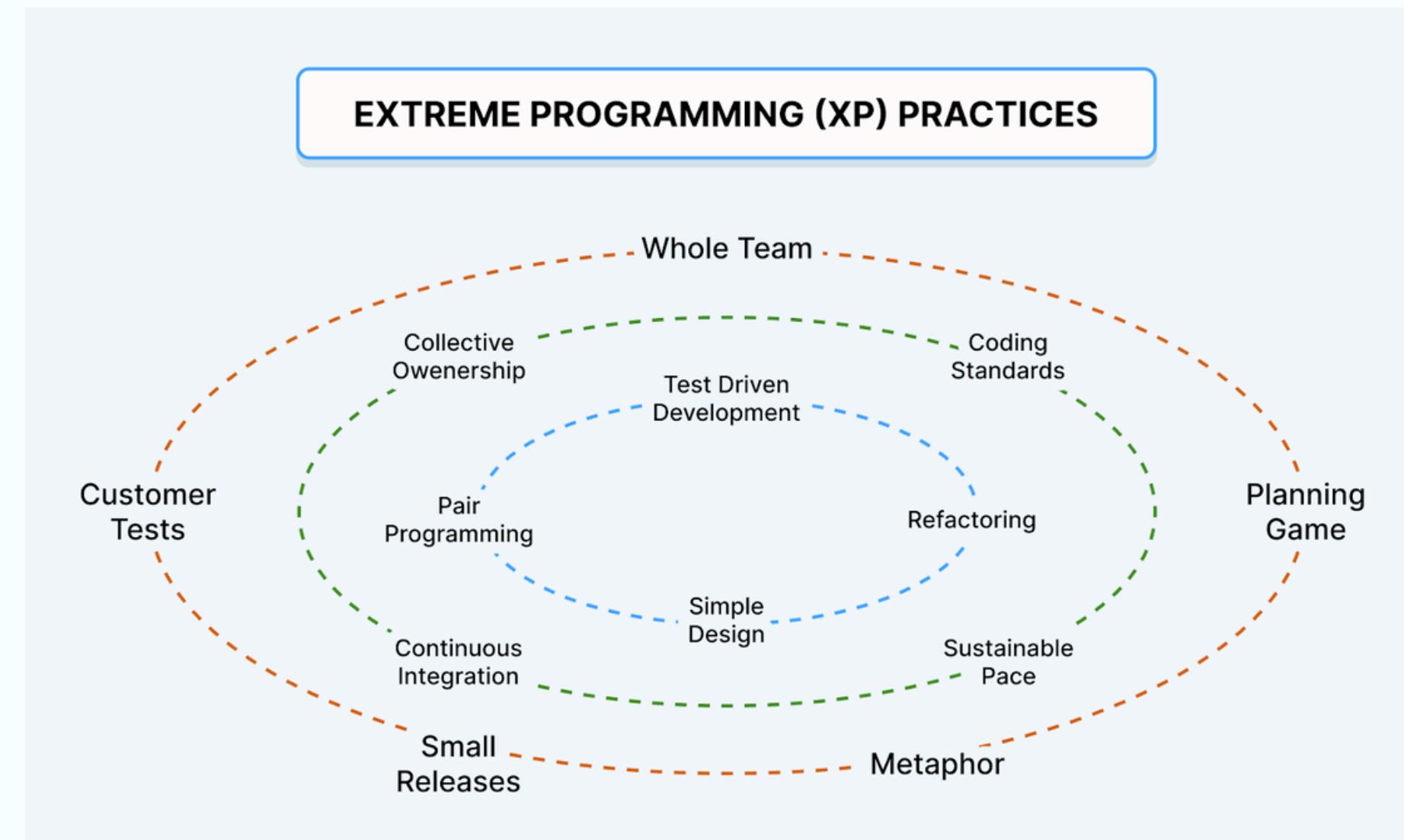


Key Principles of XP

- 1. These principles ensure smooth and effective project execution:**
- 2. Rapid Feedback:** Continuously test and review work to address issues early.
- 3. Incremental Change:** Make small, manageable updates rather than large, disruptive changes.
- 4. Quality First:** Prioritize high-quality code to reduce long-term maintenance costs.
- 5. Customer Focus:** Keep customers involved to ensure the software meets their needs.

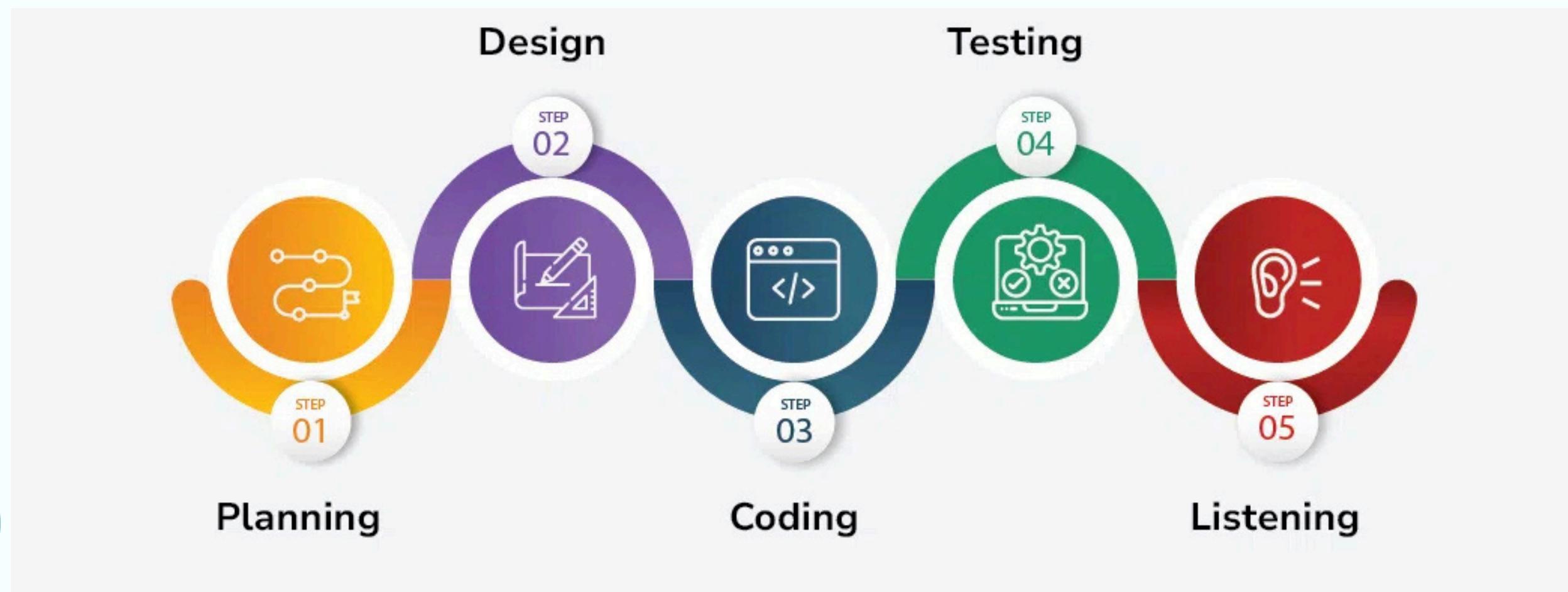
XP Practices Overview

The practices of XP are a set of specific rules and methods that distinguishes it from other methodologies. When used in conjunction, they reinforce each other, help mitigate the risks of the development process, and lead to the expected high-quality result. XP suggests using 12 practices while developing software which can be clustered into four groups.



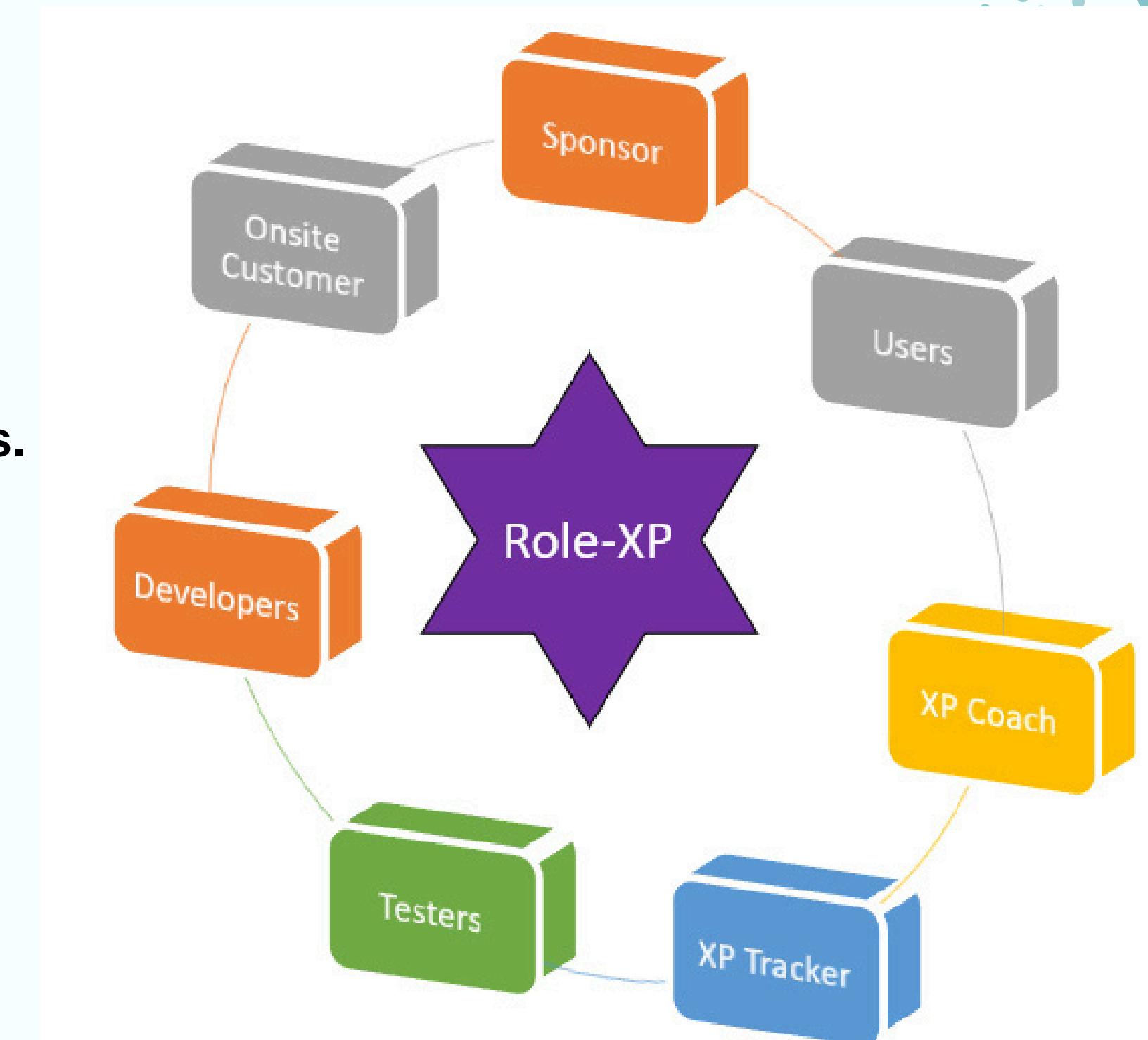
life cycle of XP

1. Exploration Phase: Gather and prioritize user stories.
2. Planning Phase: Create iteration plans and break down tasks.
3. Iteration Phase: Develop and test small increments.
4. Release Phase: Deliver functional software and gather feedback.
5. Maintenance Phase: Address defects and adapt to changes.
6. Death Phase: Conclude the project when goals are met.



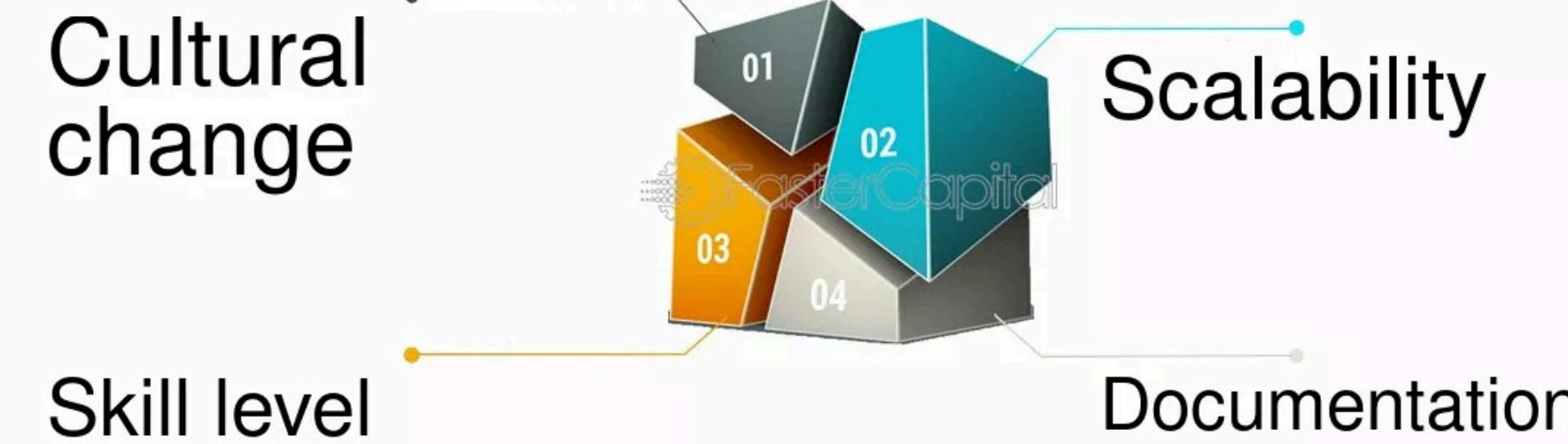
Roles in XP

- **Customer:**
Provides requirements and feedback.
- **Programmers:**
Write code and tests, implement features.
- **Coach:**
Ensures XP practices are followed.
- **Tracker:**
Monitors project progress and metrics.



Challenges of XP

- Requires disciplined and skilled teams.
- Intensive customer involvement may be challenging to maintain.
- Pair programming can be time-consuming.
- Not well-suited for large, distributed teams.



Applications of XP

- **Dynamic Environments:** Projects with frequently changing requirements, like startups or evolving industries.
- **Web and Mobile Apps:** Platforms requiring frequent updates, e.g., e-commerce or social media apps.
- **Complex Projects:** High-risk systems like healthcare or banking software.
- **Collaborative Teams:** Encourages teamwork in cross-functional or agile setups.
- **Continuous Enhancements:** Regular updates for SaaS platforms or existing software.
- **Tight Deadlines:** Quick delivery for hackathons or short-term projects.
- **Education:** Training teams in agile practices via real-world simulations.
- **Innovation Projects:** Prototypes or pilots in AI, IoT, and blockchain.
- **Agile Transition:** For teams moving from traditional to agile methods.

Advantages of 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.

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.



**Thank
You**