- **Extreme Programming (XP): Emphasizes practices such as pair programming, test-driven development (TDD), and frequent releases. It aims for high customer satisfaction and rapid iterations.**
- **Lean Software Development: Inspired by Lean manufacturing, it focuses on eliminating waste, improving efficiency, and delivering value. Practices include value stream mapping and continuous improvement.**

Comparison: Traditional vs. Agile

1. **Flexibility:**
    - **Traditional: Rigid; changes are difficult and costly once development starts.**
    - **Agile: Highly adaptable; accommodates changes at any stage of development.**
2. **Customer Involvement:**
    - **Traditional: Limited to initial phases; less interaction during development.**
    - **Agile: Continuous collaboration with the customer throughout the process.**
3. **Delivery:**
    - **Traditional: Complete product is delivered at the end of the development cycle.**
    - **Agile: Functional software is delivered incrementally with frequent releases.**
4. **Documentation:**
    - **Traditional: Extensive documentation is a priority.**
    - **Agile: Working software takes precedence over detailed documentation.**
5. **Risk Management:**
    - **Traditional: Risk management is primarily through upfront planning.**
    - **Agile: Risks are managed through iterative development and ongoing feedback.**

Conclusion

Traditional software processes provide a structured and predictable approach but can be inflexible. Agile development offers flexibility and adaptability, making it suitable for projects requiring frequent changes and customer collaboration. The choice between traditional and Agile methods depends on project needs, complexity, and the required level of responsiveness.