The Adaptive Agile project plan represents a shift from rigid, plan-driven, step-by-step methods (like traditional Waterfall) to a flexible, iterative, and customer-centric approach. By dividing the bigger abstract task into smaller business focused tasks.

1. **Iterative** and multiple smaller Deliverable:

Instead of delivering the final product at the end, work is divided into small increments (sprints). Each sprint produces a potentially shippable product. This allows for continuous feedback and course corrections.

1. **Customer Collaboration**

Agile focuses on continuous engagement with stakeholders. The project evolves based on changing needs and feedback, rather than sticking to a fixed scope.

1. **Responding to Change**

Agile project plan always accommodates new changes even late in the project. This is very useful since priorities can shift based on market conditions or customer needs.

1. **Self-Organizing Teams**

Agile teams are cross-functional and autonomous, empowered to make decisions. The project manager is also called scrum master becomes more of a facilitator rather than controlling every task.

1. **Focus on Working Software and Deliverables**

Progress is measured by smaller working solutions, not by exhaustive documentation. The emphasis is on value delivery, not just activity tracking.

**Example Project:** Implementing 5G Network Monitoring and Alerting System

The telecom landscape changes rapidly with evolving hardware, network policies, and customer needs. Stakeholders often adjust performance KPIs or introduce new thresholds based on ongoing tests. Frequent releases are required to stay aligned with real-time network performance demands.

The team breaks down the 5G monitoring system into epics like:

1. Real-time KPI dashboards
2. Custom alert rule engine
3. API integrations with OSS/BSS systems

Each sprint focuses on delivering one component, like the alert engine.

After every sprint, feedback is taken from Network operations team (NOC), Performance engineers and Business leads

Based on feedback, changes are integrated into the next sprint without needing a complete project reset. The project is continuously refined based on user needs and system data patterns.