

Development Methodology

Overview:

Praxis is equipped and well positioned to proactively adopt any of the popular development methodologies and practices for software development including Waterfall, Incremental, Agile (XP/Scrum), Spiral, Rational Unified Process (RUP), and Time boxing based on client's requirements and nature of the project.

The dynamic and competitive ambiance of today's world compels the Independent Software Vendors (ISV) to focus on quality, time-to-market, product transformation, frequent releases and product sustenance simultaneously. This has necessitated frequent espousal of Agile Methodology to manage and control product development using iterative and incremental practices. Our embracement of Agile provides the opportunity to leverage time, release quick prototypes, ensure rapid development and eliminate redundant documentation.

Praxis has successfully managed to overcome the challenges faced in working with an Agile team by combining innovative collaborative methods, adaptive processes, applications, tools and management techniques

Aspects of our Software Development Methodology:

- * Make 100% test coverage easier to attain
- * Incremental addition of complexity
- * Progress rate can be measured early in the process
- * Delivery of working software is repeated several times within a release
- * Progress can be measured without requiring faith in the method itself
- * Shorter cycles force engineers to automate builds, unit tests and deployments
- * Repeating the same process several times within the release cycle increases accuracy
- * Product requirements and design processes are more intense and detailed
- * Design and code review processes utilize more automation and asynchronous steps
- * Product release processes are included to bring quality products to market - not only applications to users
- * Product maintenance, support processes are included to cover the complete Software Performance Engineering (SPE)

lifecycle

- * Simple implementation due to wrapping existing engineering practices (Extreme Programming, RUP)
- * 80-20 Agile-Waterfall infrastructure
- * Clearly documented work practices
- * Simple but effective tools