**Development & Testing with Agile: Extreme Programming**

**What is Extreme Programming (XP)?**

Extreme Programming (XP) is an agile software development framework that aims to produce higher quality software, and higher quality of life for the development team. XP is the most specific of the agile frameworks regarding appropriate engineering practices for software development.

**The general characteristics where XP is appropriate are**

**\*** Dynamically changing software requirements

\* Risks caused by fixed time projects using new technology

\* Small, co-located extended development team

\* The technology you are using allows for automated unit and functional tests

**AGILE TESTING**

Agile Testing is quite different from traditional testing methods of sequential processes. Requirements and documentations are not necessary for test processes in agile. Testers are allowed to join the developers and users in their initial plan meetings. During the meeting tester himself notes down the requirements and then matches the developed codes to those requirements

**EXTREME PROGRAMMING**

Extreme programming (XP) is a well known agile practice. XP tends to manage project tasks, project manuals and documentation to reduce cost and support changing requirements. XP is based on an iterative and incremental approach normally executed in small cycles. Xp brings whole team together thereby increasing productivity and creating a cohesive working environment. Idea behind teamclustering is to get enough feedbacks, so that team can highlight their weak points and tune them accordingly. This is why XP is normally said to be a people oriented process rather than process oriented .

Xp has its various dialects. Here focus of interest would be

1) Pair programming

2) Ping Pong Programming

**Pair Programming:**

Pair programming is a concept used within XP, In pair programming, two people work in collaboration on the same algorithm, code or designing task. The two involved participants sit at one working computer side by side. One of the people is responsible for writing the code or designing the algorithm while the other person sits beside him and keep reviewing the coding. The first person who is coder is generally known as “Programmer” while the other one who reviews the code is known as “navigator” or “observer”. This practice improves the software development process.

**1)Code reviews**

In Code Reviews, Reviewers sit down by taking a dedicated time out to review someone’s code, everyone of these reviewers have some comments related to the code but not all of them will work on that code on daily bases, Everyone seems to be engaged and involved in discussion at that time but once review meeting is done not all of them will actually be a part of coding process, which eliminate the effective feedback loop.

**2)** How code reviews are related to pair programming Code reviews are done to analyze the quality of code whereas pair programming is designed to produce a reviewed code.

In pair programming, Driver and Navigator are continuously writing and reviewing the code thus increasing feedback loop effectiveness, which eliminates the code review overhead in later stages. Introducing pair programming and its variants in agile software development and testing increases the code qualities which ultimately decrease the chances of code review meetings in later testing phase.

**2. Ping Pong Programming**

Both members of the pair write code. One of them writes a test code and the other writes the actual code(product code). For example A writes a test code for product code which is written by B.A aims to produce a failing test code whereas B aims to pass that test code.

Tested code Ping pong approach is designed to produce a tested code. There is a difference between tested and reviewed code. A reviewed code is such a code which is clean and is according to reviewed code standards. Tested code is such a code which actually passes all necessary test cases. Any code which is tested code does not necessarily assure that the code is clean and reviewed.