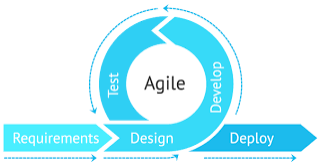
Software Engineering And Testing

Agile Development



Outline

- Agility and Cost of Change
- Agile Process
- Introduction to Agile Process Models:
  - Extreme Programming
  - Adaptive Software Development
  - Dynamic Systems Development Method
  - Scrum
  - Crystal
  - Feature Driven Development
  - Lean Software Development
  - Agile Modelling
- Agile Unified Process
- Advantages and Disadvantages of Agile.

Agility

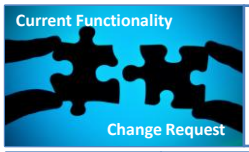
Agility is **ability to move quickly and easily**.

It is a property consisting of **quickness**, **lightness**, & **ease of movement**.

- The ability to **create** and **respond to change** in order to profit in a unstable global business environment.
- The ability to **quickly reprioritize use of resources** when requirements, technology, and knowledge shift.
- A very **fast response to sudden** market **changes** and emerging threats by intensive customer interaction.
- Use of **evolutionary**, **incremental**, and **iterative delivery** to converge on an optimal customer solution.
- Maximizing **BUSINESS VALUE** with **right sized**, **just- enough**, and **just-in-time processes** and **documentation**.

3

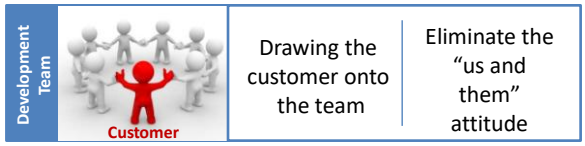
What is Agility?



Effective response to change

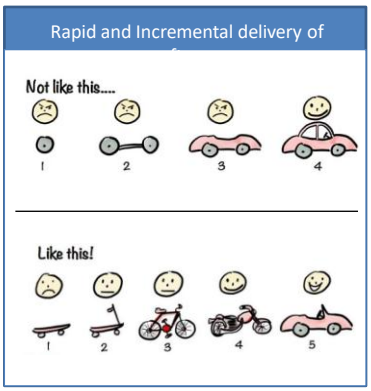
Effective communication among all stakeholders

Organizing a team so that it is in control to perform the work



Drawing the customer onto the team

Eliminate the "us and them" attitude



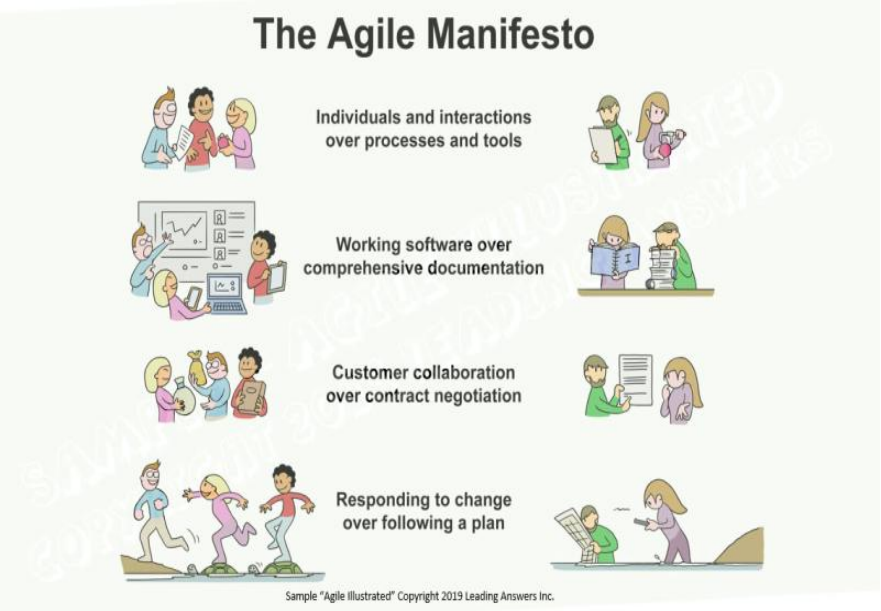
4

Agile Process

- Agile software process addresses **few assumptions**
  - **Difficulty in predicting changes** of requirements and customer priorities.
  - For many types of software; **design** and **construction** are **interleaved** (mixed).
  - **Analysis, design, construction** and **testing** are **not** as **predictable** as we might like.
- An agile **process** must **adapt** incrementally.
- To accomplish incremental adaptation, an agile team **requires customer feedback** (so that the appropriate adaptations can be made).

5

Agile Manifesto



Agile Principles

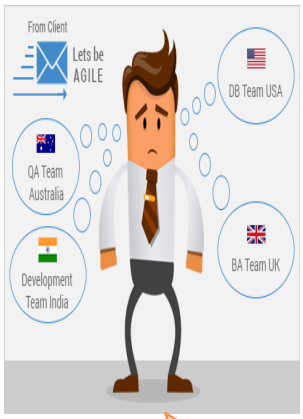
- **Highest priority** is to **satisfy** the **customer** through early & **continuous delivery** of **software**
- **Welcome changing** requirements
- **Deliver** working **software frequently**
- **Business people** and **developers** must **work together**
- **Build** projects **around motivated** individuals
- Emphasize **face-to-face conversation**
- **Working software** is the **measure** of **progress**
- Continuous **attention** to **technical excellence** and **good design**
- **Simplicity** – the art of maximizing the amount of work done
- The best designs emerge from **self-organizing teams**
- The **team tunes** and **adjusts** its **behaviour** to become more effective
- Agile process promote **sustainable development**.

7

Where Agile methodology not work

Project plan & requirements are clear & unlikely to change

Unclear understanding of Agile Approach among Teams



Big Enterprises where team collaboration is tough

8

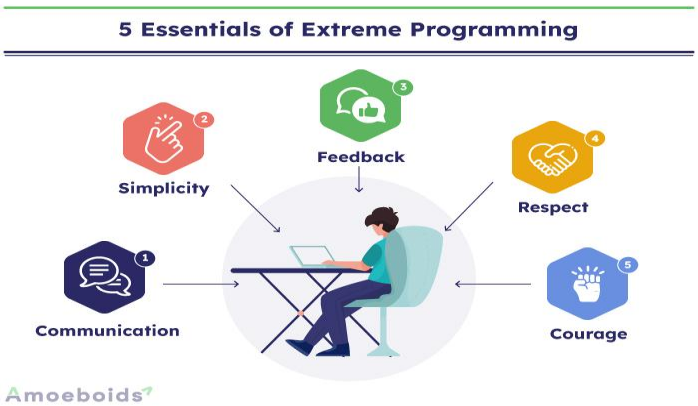
Agile Process Models

- Extreme Programming (XP)
- Adaptive Software Development (ASD)
- Dynamic Systems Development Method (DSDM)
- Feature Driven Development (FDD)
- Crystal
- Agile Modelling (AM)

9

Extreme Programming

- The most widely used approach to agile software development
- A variant of XP called **Industrial XP (IXP)** has been proposed to target process for large organizations
- It uses **object oriented approach** as its preferred development model



10

Extreme Programming

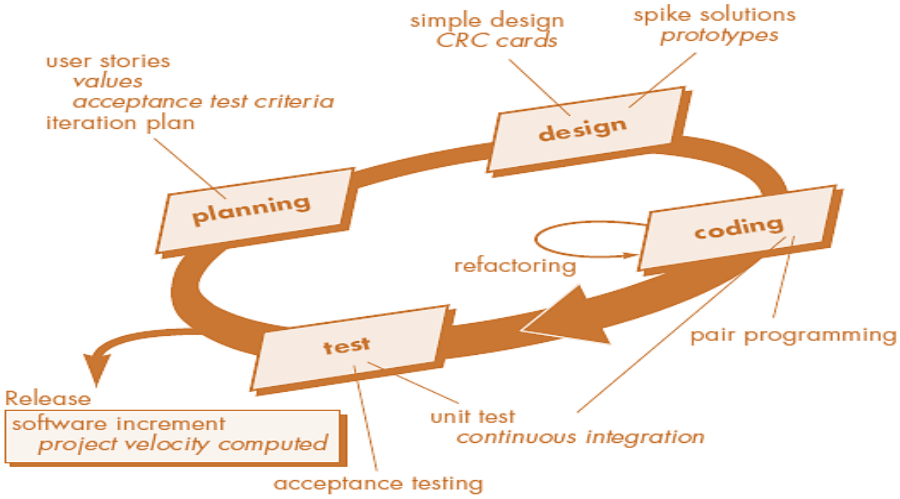
XP Values

- **Communication:** To achieve effective communication, it **emphasized close & informal (verbal) collaboration** between customers and developers
- **Simplicity:** It restricts developers to **design** for **immediate needs not** for **future needs**
- **Feedback:** It is derived **from** three sources the **implemented software**, the **customer** and **other software team members**, it uses **Unit testing** as primary testing
- **Courage:** It demands courage (discipline), there is often significant pressure to design for future requirements, XP team **must have the discipline (courage) to design for today**
- **Respect:** XP team **respect** among **members**

11

The XP Process

- It considers four framework activities
- **1.** Planning ◆ **2.** Design ◆ **3.** Coding ◆ **4.** Testing

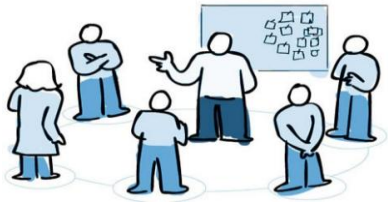


12

The XP Process

Planning

- User Stories
  - **Customers assigns value** (priority)
  - **Developers assigns cost** (number of development weeks)
- Project velocity
  - Computed at the end of first release
  - **Number of stories implemented in first release**
  - Estimates for future release
  - **Guard against over-commitment**



The XP Process

Design

CRC card

Class Name	
Responsibilities	Collaborators

- **Keep-it-Simple** (Design of extra functionality is discouraged)
- **Preparation of CRC card** is work project
  - CRC cards identify and organize object oriented classes
- **Spike Solutions (**in case of difficult design problem is encountered**)**
  - Operational prototype intended to clear confusion
- Refactoring
- Modify internals of code, No observable change

Coding

- **Develops** a series of **Unit test** for stories included in current release
- **Complete code** perform **unit-test** to get immediate feedback
- XP recommend **pair-programming**, **"Two heads are better than one"**
- **Integrate code** with other team members, this **"continuous integration"** helps to avoid compatibility & interfacing problems, **"smoke testing"** environment to uncover errors early

Testing

- **Unit test** by **developers** & fix small problems
- **Acceptance tests** - Specified by **customer**
- This encourages regression testing strategy whenever code is modified

14

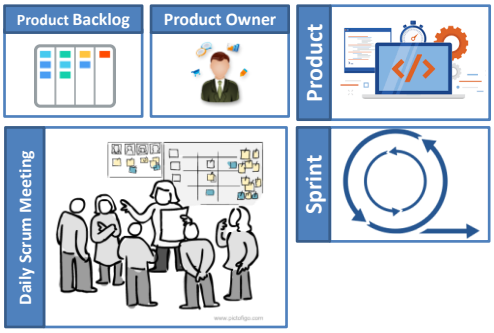
What is Scrum?

Scrum is an agile process model which is used for **developing** the **complex software** systems.

A scrum is a method of restarting play in rugby that involves players packing closely together with their heads down and attempting to gain possession of the ball.

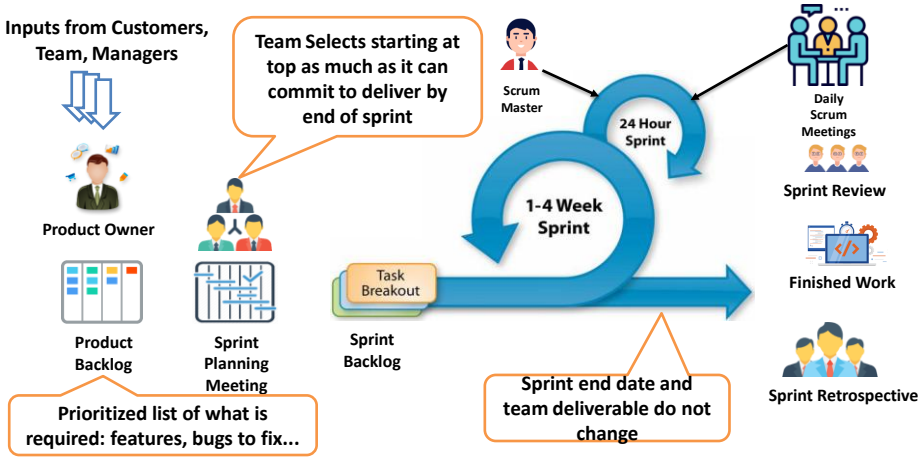
It is a **lightweight process framework**.

Lightweight means the **overhead of the process is kept as small** as possible in order to maximize the productivity.



15

Scrum framework at a glance



16

Scrum

Backlog

- It is a **prioritized list of project requirements** or features that must be provided to the customer.
- The **items can be included** in the backlog at **any time**.
- The **product manager analyses** this **list** and **updates** the **priorities** as per the requirements.

Sprint

- These are the **work units** that are needed **to achieve** the requirements mentioned in the backlogs.
- Typically the sprints have **fixed duration** or time box (of **2 to 4 weeks, 30 days**).
- **Change** are **not introduced** during the **sprint**.
- Thus sprints allow the team **members** to **work** in **stable** and **short-term environment**

17

Scrum

Scrum Meetings

- There are **15 minutes daily meetings** to **report** the **completed** activities, **obstacles** and **plan** for **next** activities.
- Following are three questions that are mainly discussed during the meetings.
  - **What** are the **tasks done** since **last meeting ?**
  - **What** are the **issues** that team is **facing ?**
  - **What** are the **next activities** that are **planned?**
- The **scrum master** leads the meeting and **analyses the response** of each team member.
- Scrum meeting **helps** the **team** to **uncover potential problems** as early as possible
- It leads to **"knowledge socialization"** & promotes **"self-organizing team structure"**

18

Scrum

Demo

- Deliver **software increment** to customer
- Implemented functionalities are **demonstrated** to the customer

19

Practices	*Scrum*	*Kanban*	*XP*	*DSDM*	*FDD*	*Crystal*
Approach	Iterative Increments	Short Iterations	Increments are Iterative	Iterative	Iterative	Incremental/Iterative
Time	2-4 weeks	Continuous Delivery	1-6 weeks	80% solution in 20% time	Two days to 2 weeks	Frequent Delivery
Team Size	5-9	Small to medium	Small team, 2-10	2-10, independent teams	4-20, more than one team	Starts from as low as 6 to larger teams
Suitable Project size	All types	All types	Smaller projects	All types	Large	Small and medium scale project
Major Practices	Sprint, Product Backlog, Sprint Backlog, Scrum meetings	Kanban board, stickies	User stories, refactoring, pair programming	Prototyping, Feasibility and business study	UML diagrams	Crystal Family, Criticality

Advantages of Agile Methodology

- Agile is very much suited for projects where requirements and the end product is not very clear.
- It promotes customer satisfaction as their feedbacks and changes are embraced.
- It reduces risk factors as early deliverables are made visible to the end-users.
- Exhaustive planning is not required at the beginning of the development process.
- It is easy to manage with minimal rules and more flexibility.
- Dividing the project into incremental deliverable builds leads to more focus on the quality of the product.

21

Disadvantages of Agile Methodology

- As it is highly customer-centric, so it can pose a problem when the customer does not have a clear understanding of the product and process.
- Lack of formal documentation and designing leads to a very high dependency on individuals for training and other tasks.
- For complex projects, the resource requirement and effort are difficult to estimate.
- Frequent deliverables, feedback, and collaboration can be very demanding for some customers.
- Because of the ever-evolving features, there is always a risk of the ever-lasting project.

22


THANKS