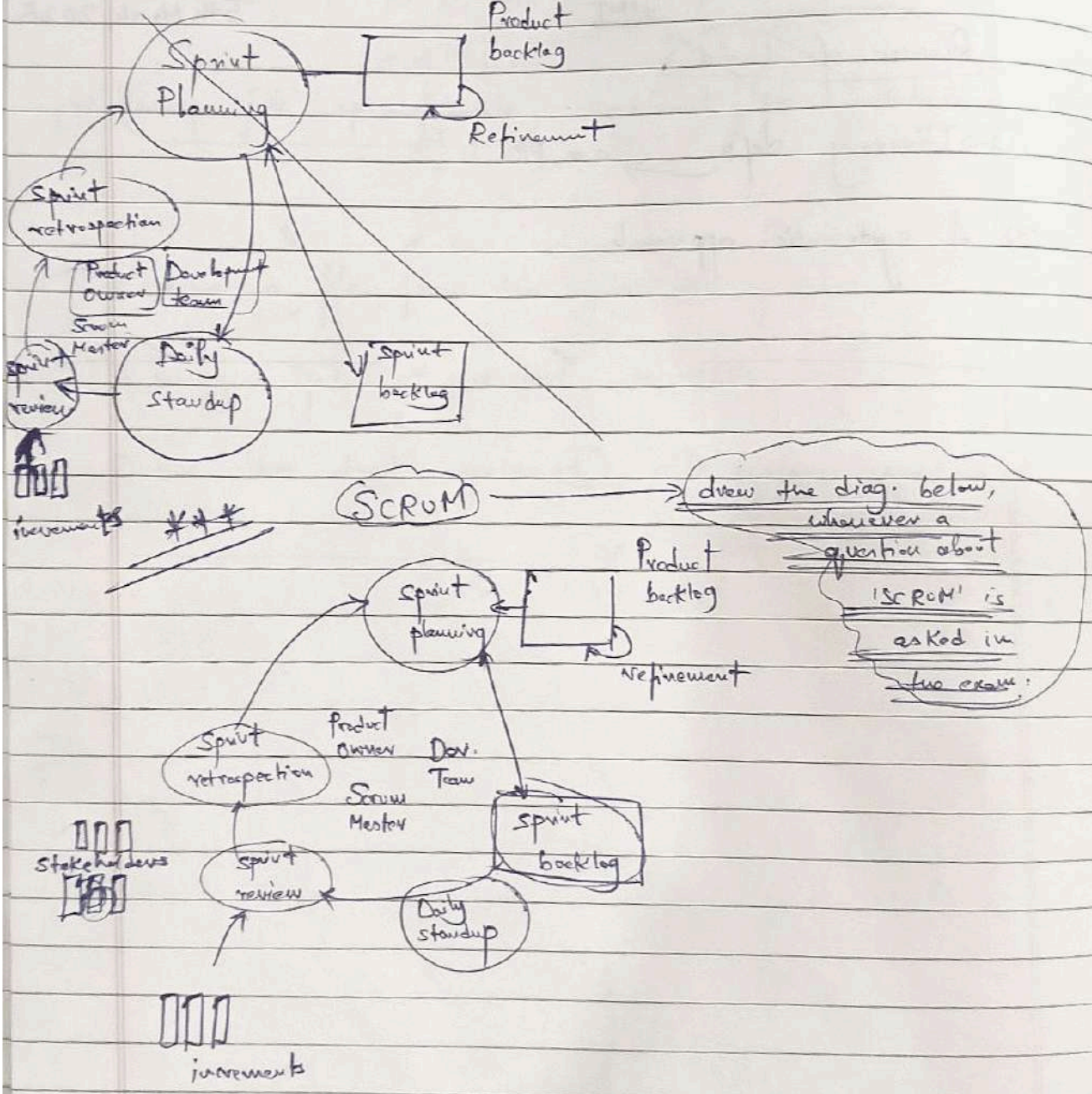
Page :
Date: / /

AGILE

5th March, 2025



Page :
Date : / /

Role of Product Manager

(1) → has pivotal role

(2) → effective comm. & resolving conflicts

Conflict Resolution Techniques

- → open dialogues
- → Negotiation
- → collaborative problem solving
- → de-escalation

→ clear expectations

→ active listening

→ transparent communication

listen actively to the team & to the stakeholders. Hence, ensuring the issue is addressed promptly.

(1) implemented through daily standup, sprint review, sprint retro.

(2) Regular updates (standup meeting, sprint reviews; make the team stay aligned & deliberate openly & honestly on the progress, blockages, and changes.

Early Conflict Identification

identifying the conflict early, by seeing the team dynamics.

GPT

goal

timelines

priorities

Team based exercises to simulate cross functional collab.

* see next page

Page :
Date: / /

* Exercises :-

1) Paper Tape Tower (creation)
2) Customer Journey Mapping
3) Feature prioritization
4) Product Demo Simulation
5) Rapid Prototype Challenge
6) Go to market strategy.
7) Agile lego game.

Decision Capability + stake holder management

Page:
Date: / /

12th March

AGILE METH

Unit 3

Agil

AGILE SOFTWARE DEVELOPMENT → FCR → Flexibility, Collaboration, Delivering Value

→ Why?

- Core Fundamentals of AGILE → Iterative development, customer feedback, flexibility
- Role of Product Manager → roles & responsibilities → Vision, Backlog Mgmt, Customer Advocacy

There are phases in this are:-
- → concept
- → inception
- → iterations
- → release
- → sprint review/ sprint retrospective

→ AGILE S/W DEVELOPMENT LIFECYCLE
→ AGILE ROADMAP
→ Prioritization of Backlog (the techniques used) → MoSCoW, → WSJF

Customer Centric Development

- → Zara
- → Spotify
- → Netflix
- .
- .
- .
- .

Page :
Date: / /

Roadmap Development in Agile

- → Vision
- → Mission
- → Themes & Epics
- → Product & Features
- → Release Planning
- → Backlog Management
- → Timelines.

example → Q2. 2025 : improving user experience & background scalability

→ the mission can be divided into themes, which shall be covering diff. aspects related to the Mission.

eg: Epics Theme1: Improving User Exp.

related epics → epic 1, epic 2, ...

Theme2: Improving & scalability.

epic 1
epic 2
...

contains

Theme is a group of related epics.

for long answer questions:

→ viz: Roadmap, eg., components

→ agile manifesto, agile principles, wi

→ techniques in unit 3 wi

Page :
Date : / /

Prioritization Techniques for Backlog

MoSCoW method

- also used for classification of tasks and features based on their importance.

M (Must-have) — ~~essential features required for the product to have function.~~ critical features.

S (Should-have) — important but not critical features

C (Could-have) — can be there, but not ~~contributing~~ among the core features.

W (Won't-have) — the features that are out of scope for current iteration but may be considered later.

WSJF

Weighted Shortest Job First

Cost of Delay
- User experience
- time criticality
- Risk reduction
- opportunity enablement

Page :
Date : / /

AGILE

Chapter : 1

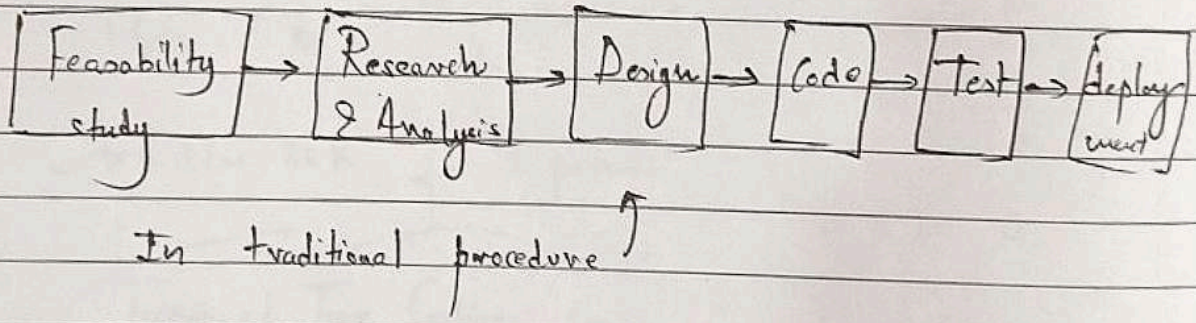
Beginning: TDD — BDD, [illegible], P. Pair ..., CI, ...

TDD ↓

Test Driven Development

2nd point → Agile Metrics

cycle time, velocity, sprint burndown chart
sprint burnup chart, Lead Time ...



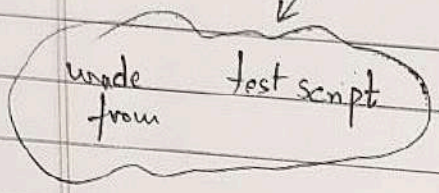
In traditional procedure ↑

Testing:

→ Actual o/p
→ Expected o/p

→ if actual o/p does not match with the expected o/p, then there exists error.

Testing → Test Suite : collection of test suites cases



(I, S, R)

input ... ... (see ppt.)
(see ppt)

TDD Test Driven Development

(1) Red → Write a failing test.

Green → Write minimal code for test to pass.

Refactor → Optimize the code for enhancement, without changing external behaviour.

(2) In TDD

① → test-cases are written before writing actual code.

② → the Read-Green-Refactor cycle is followed.

Page :
Date: / /

advantages

(4) pros of using TDD:

→ reduces de-bugging time
→ encourages modularity.

(3) - refer the PPT for example on TDD (explain using login page example)

BDD → Behaviour Driven Development

(1) → D C T and Non-technical member can understand, as things are written in natural language (like English)

Developer Coder Tester

(2) Acceptance Test Criteria → written in Gherkins → the scenarios are written & described in English.

Given
When
Then

eg:

Feature: Shopping Cart
Scenario: adding an element to the shopping cart

Page:
Date: / /

(3) Give example to explain the entire BDD thing.

↓

see from PPT.

Pair Programming

→ Two people working

Type I

① Driver
② Navigator

advantage of pair programming
① → shared knowledge
② → collaborative

→ example of recursion code to calculate factorial of number.
simple

explain with this simple general example.

Type II

Ping-Pong

→ roles alternate

Type III

Test driven programming

Strong typed programming

Active-Passive Programming

Page :
Date: / /

Type IV — Remote Pairing

Type V — Idle Pairing

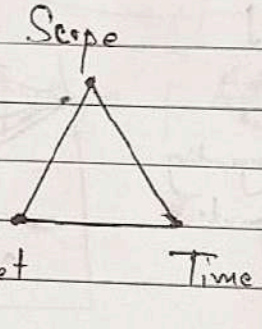
Page :
Date : / /

AGILE

09th April, 2025

Agile · Metrics

Project
→ Velocity → indicator of the work done. capable to that is capable to be done by an AGILE team in a future project.
→ Sprint Burndown Chart.
→ Sprint Burnup Chart.
→ Cycle Time
→ Lead Time (Lead is in '+', Lag is in '-')



example:

PROJ 1

SPRINT 1 – 30 story points
SPRINT 2 – 40 "
SPRINT 3 – 50 "

$\frac{30+40+50}{3}$ = 40 story pts.

next proj → (lag)

PROJ 2

SPRINT 1
SPRINT 2
SPRINT 3

→ 40 story pts. will be there as the least no. of

Say in PROJ 3

the least no. of story pts. drops down to 20 (say)

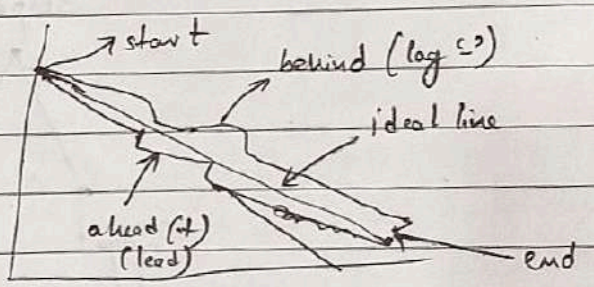
this means rate of progress is going down

So, velocity is an indicator of progress.

SPRINT BURNDOWN CHART

there exists burndown chart for the entire project also.

this is just regarding sprint.



components

→ starting pt.

→ ending pt.

→ ideal work line

→ actual work line

Page:
Date: / /

AGILE

Chapter 5 - Scaling Agile Frameworks

~~SAFe (Scaled Agile Framework)~~

Challenges in case of Small Teams

A C D

- A → Alignment
- C → Consistency
- D → Faster delivery

ensures → Collaboration & flexibility

To ensure collaboration & flexibility in larger teams :-

1. → Safe (Scaled Agile Framework) → 8-12 weeks
2. → LeSS (Large Scale Scrum) → scaling up → don't make major changes
3. → SCRUM@SCALE
   - → Agile product owner
   - → Scrum master
   - → Development team

SCRUMOFSCRUM
MetaSCRUM

SCRUMOFSCRUM
MetaSCRUM

SCRUM → Sprint → 2-4 weeks → ART's & RTE's → PI's

ART's → Agile Release Trains
RTE's → Release Train Engineers

SCALED AGILE

ROLES
- → Product Owner
- → Product Manager
- → Release Train Engineer
- → Business Owner (Technical Directions)

(SCRUMOFSCRUM)

(METASCRUM)

Confluence

→ for documentation in agile.

⇒ Asynchronous Communication

⇒ Asynchronous Communication

- Daily Standup / Daily Scrum → SCRUMOFSCRUM
  changes to

SCRUMOFSCRUM and MetaSCRUM are many a time, used interchangably in the websites & on the INTERNET, but they are not same.

Page :
Date : / /

Best Practices for Distributed Agile Teams

1) → Freq. Communication
   → Daily Standup (RTE is)

2) → Timezone & Calendar Management

3) → Collaboration Tools → version control tool
   → task mgmt tool → Jira

4) → Cross-functional teams

AGILE

SaFe → program backlog is there.

program backlog
↓
longer program backlog
↓
portfolio

① → finely grained
② → coarsely gained

finely grained is better than coarsely grained.

→ Lean Portfolio Mgmt. - aligns strategy & execution by applying lean ...

AGILE

Pointers to add while writing on Safe

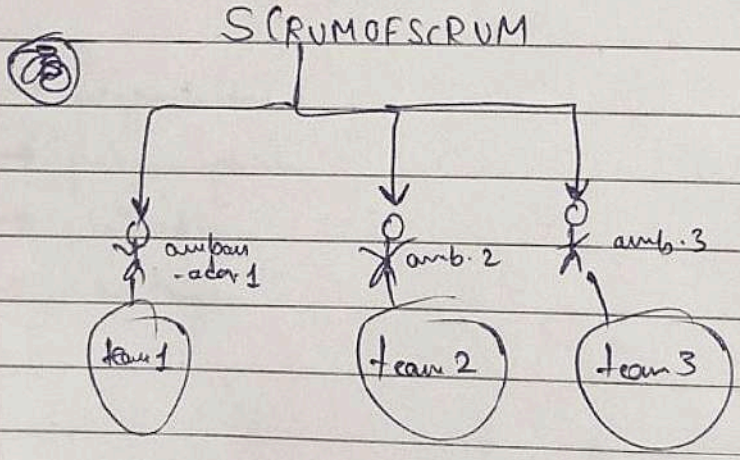
① Agile Teams

Agile Sub team 1 | Agile Sub team 2 | ... | Agile Sub team n.

②

SCRUM ROLES	SaFe
PO	PO
SM	PM
DT	RTE
	System Architect

③ SCRUMOFSCRUM



→ P.O. and P.M. are not the same in SCRUMOFSCRUM.

→ SCRUMOFSCRUM and metaSCRUM though are interchangably used, are not the same.

Page:
Date: / /

Just, Recapping:

SCRUM
→ for smaller teams

for larger teams, we:-

① → SaFe (Scaled Agile Framework)
② → LeSS (Large Scale Framework)
③ → SCRUM@SCALE

Pointers to add while writing on LeSS:

B.

- LeSS is simple, lightweight framework that scales SCRUM to multiple teams on a single product.

Pointers to add while writing on SCRUM@SCALE:

Basic SCRUM

- SCRUM Master Cycle — focuses on process
- Product Owner Cycle — focuses on delivering value.

Date: / /

- talk about SCRUMOFSCRUM and ~~MetaSCRUM~~ metaSCRUM ~~in~~ here.

→ if SCRUM master is involved into the meetings, and ~~player~~ is majorly involved ~~pivotal~~ role → ~~its~~ its SCRUMOFSCRUM

→ if there is involvement of product owner → its metaSCRUM.

7 Best Practices for Distributed Agile Team

1) Communication
   - → daily standup
   - → tools for communication
   - → asynchronous communication

2) collaborative tools
   - versioned control
   - task mgmt.
   - ~~cross functional team implementation~~
   - ~~timezone & calendar management~~

3) Cross Functional Teams

4) Timezone & Calender Mgmt.

Page:
Date: / /

Agile

Challenges faced whilst scaling agile across distributed large teams.

- → Communication & Coordination across teams
- → Managing Dependencies
- → Maintaining Consistency
- → Tooling Complexity
- → Resource Management

Page:
Date: / /

AGILE

Extreme Programming (XP)

is followed in Agile

KIS ← Agile → collab.

keep it simple

A C R → faster delivery.

alignment

Values of XP:

1. → communication
2. → simplicity
3. → feedback
4. → courage
5. → respect — respect each other's opinion.

Twelve Key Practices of XP

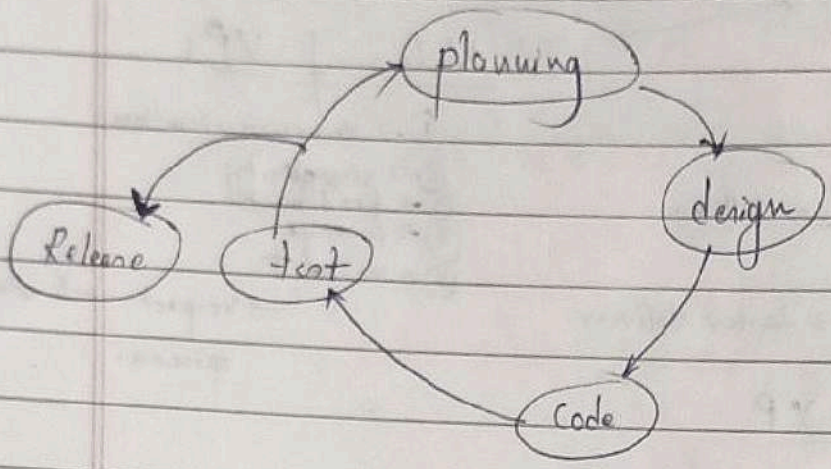
1. Planning Game
2. Small Releases
3. Metaphor.
4. Simple Design
5. Testing
6. Refactoring
7. Pair Programming
8. Collective Code Ownership
9. Continuous Integration
10. 40-Hour Week
11. On-site Customer
12. Coding Standards.

→ See PPT . . .

→ RAD (Rapid Application Development)

AOP (Aspect Oriented Programming)

XP



→ XP follows Object Oriented Approach