

Reasons for s/w failures,

- Time exceeds
- Cost over run
- Does not meet the user requirements
- Less quality

What is agile?

- A philosophy with set of guidelines for s/w development
- An Agile s/w development is a set of methods which follows the agile manifesto

Agile Manifesto

- Individuals & interactions over process & tools
- Customer collaboration over contract negotiation
- Responding or welcoming to the changes over following a plan
- Working s/w over comprehensive documentation

Agile methods - suitable for ?

- small / medium sized product devel.
- custom system development in organi.
- innovative new product / enhancing existing products
- experiments - critical system engineer,

Principles Behind Agile Manifesto

- (1) Highest priority is to satisfy the customer
- (2) Welcome changing requirements
- (3) Deliver working software frequently
- (4) Business people and developers must work together
- (5) Build projects around motivated individuals
- (6) face to face conversation
- (7) Working software is the primary measure of progress
- (8) Promote sustainable development
- (9) Technical excellence and good design
- (10) Simplicity
- (11) Self organizing teams
- (12) Team reflection

Features of agile development

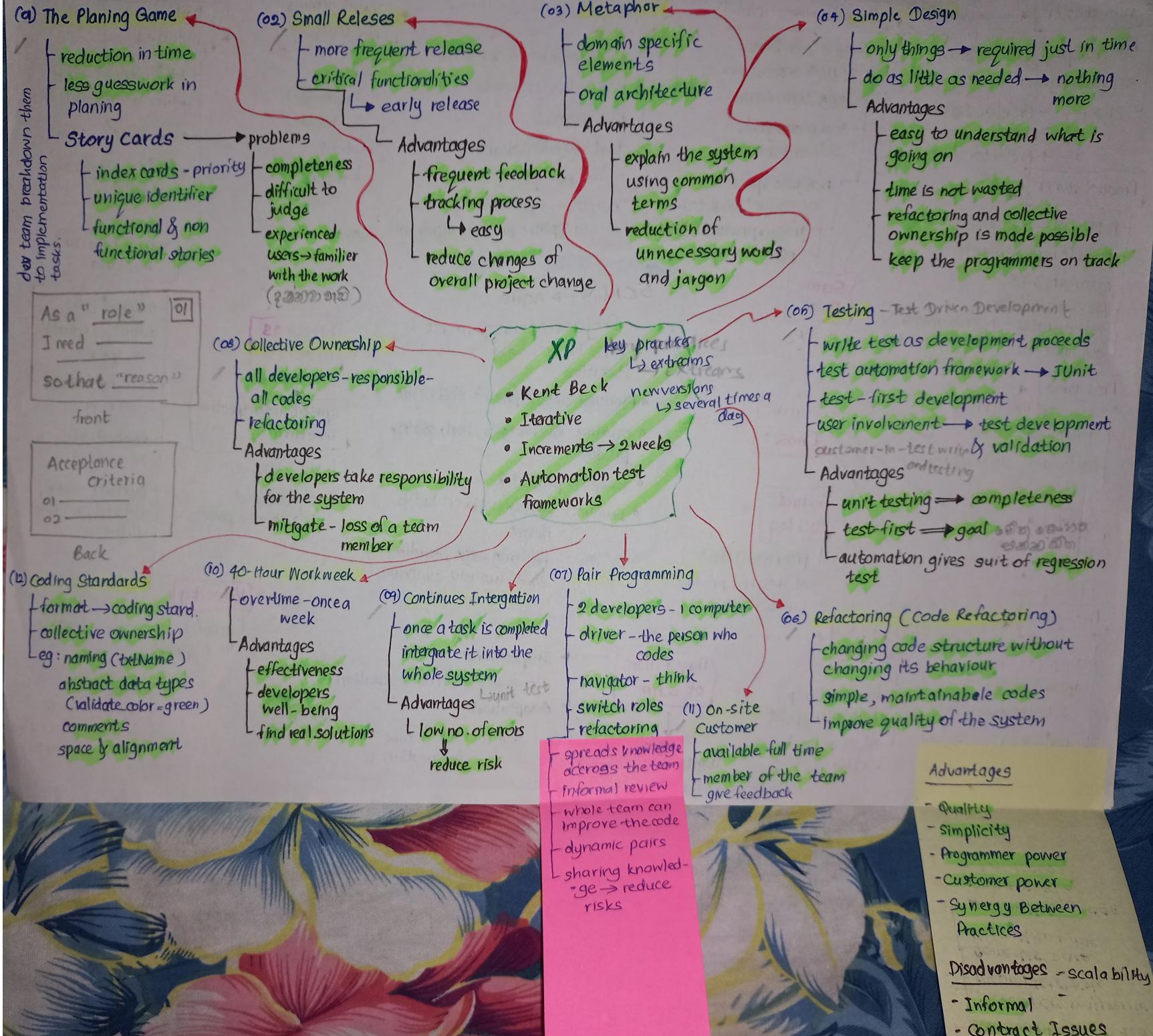
- specification, design & implementation are interleaved
- increments
- frequent delivery
- Extensive tool support
- minimal documentation
- informal technique

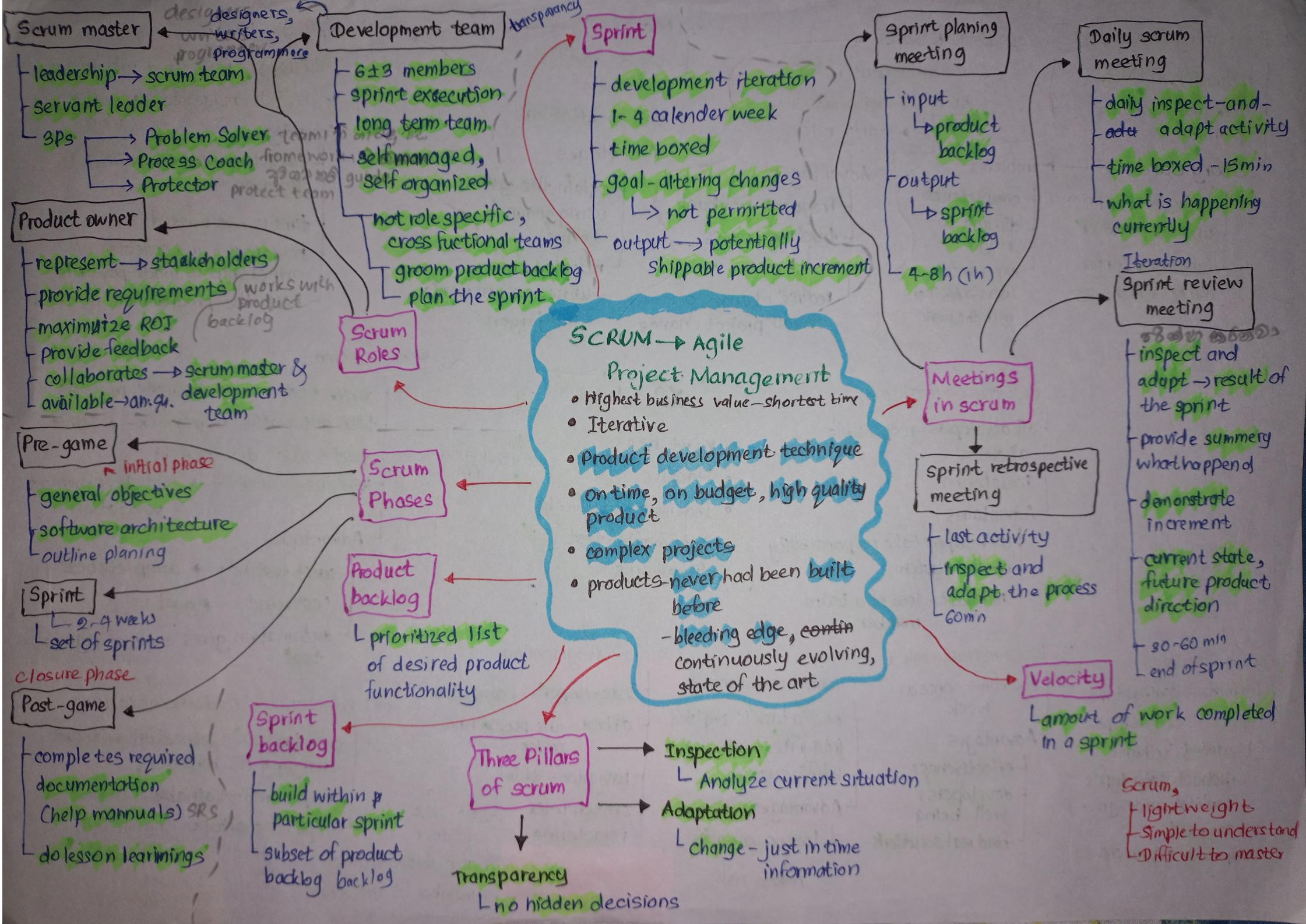
Agile Development Techniques

- XP
- Scrum
- Lean s/w development
- Kanban s/w development
- Feature driven development (FDD)
- Dynamic System Development Methods (DSDM)
- Crystal Family

Drawbacks

- great team work required
- each individual responsible for small part of work
- great deal of coordination or communication is required
- prioritizing changes can be difficult
- team may not have time to carry system implications
- team members may unlike to change current practices.





Product owner

- balance the needs of stakeholders
- setting clear direction / vision
- manage product backlog
- product owner doesn't create all the requirements

Development Team

- designers, writers, programmers etc.
- deliver work through the sprint
- ensure transparency
- daily standup meeting
- self managed, self organized
- not role specific but cross functional

Scrum Master

- responsible for holding everything
- ensure scrum is being done well.
- problem solver
- transparency : scrum team - trans
- empiricism : coach scrum team (breaking down work ...)

Scrum Benefits

- project is broken down into manageable and understandable chunks
- unstable requirements don't hold up progress
- team - visibility of everything
- frequent customer feedback
- trust between customers and developers
- positive culture

SCRUM

- self organization : try different values
 - courage, focus
 - commitment
 - respect
 - openness
- doesn't run daily scrum meeting

KANBAN

- * framework used to implement agile s/w development
- * real time communication
- * full transparency
- * work items are represented visually on a kanban board
- * visualize the work limit, work-in-progress (WIP) and quickly move work from "Doing" to "Done".
- * Allow team members to see the state

Kanban

- No defined roles
- Continuous
- Changes can be incorporated any time
- Kanban board
- Tools - Jira s/w, kanbanize, Swift-Kanban, Trello, Asana
- Effective, efficient predictable

Scrum

- Scrum master, product owner, development team
- Sprint cycle lasts 1 - 4 weeks
- changes are not made during the sprint
- product backlog, sprint backlog, product increments
- Tools - Jira s/w, Axsoft, VivifyScrum, Targetprocess
- Transparency, adaption, inspection.