

Pages 3

Roll No - 170102081

Date - 15/10/20

Agile

- ① The agile model does not recommend developing prototypes but emphasises the systematic development of each incremental feature at the end of each iteration.
- ② Break down the solution into features that are incrementally developed.

Agile

- ① Main principle: achieve agility by removing unnecessary activities that waste time.
- ② Focus on the delivery of an incremental to the customers after each time box.
- ③ Customer interaction is more frequent and don't rely on documentation.

RAD

- ① The centre frame of RAD is based on designing quick & dirty prototypes which are then refined into production quality code.
- ② Focus on developing all the features by first doing it badly and then successively improving the code.

Spiral

- ① Main principle is risk handling.
- ② Deals with various kinds of unanticipated risks.
- ③ Less frequent, proper documentation is required.

Advantages

1. Very realistic approach to software development.
2. Promotes team work and error training.
3. Functionality can be developed rapidly and demonstrated.
4. Focus on users. It uses stories with business, focused acceptance criteria to define product features.
5. Transparency: It provides unique ~~opportor~~ opportunities for clients or customers involved throughout project.
6. Easy to manage.
7. Allows for changes. There is an opportunity to constantly refine and ~~reprioritise~~ ~~re-prioritise~~ reprioritise the overall product backlog.
8. Give flexibility to developers.
9. Minimal rules: Documentation easily employed.

Disadvantages

1. Less predictability: For some software deliverables, developers can't quantify the full extent of required efforts. This is especially true in beginning of the life cycle or large product.
2. Lack of necessary documentation: because requirements for software are clarified just in time for development, documentation is less detailed.
3. Not suitable for handling complex dependencies.
4. An overall plan, an agile leader & agile project ~~man~~ manager practice is a must without which it will not work.

5. Transfer of technology to new team members may be quite challenging due to lack of documentation.
6. Heavy dependency on customer interaction so if the customer is not clear, team maybe diverted to wrong direction.
7. More risk of sustainability, maintainability and extensibility.
8. Strict delivery management dictates - the scope, functionality to be delivered and adjustments to meet the deadlines.