**CHAPTER-2**

**PROJECT MANAGEMENT**

**2.1 PROJECT PLANNING**

**2.1.1 Project Development Approach and Justification (Process Model Used)**

One of the biggest factors that dictate your choice of a life cycle method is the clarity and stability of the project requirements.

**Why Scrum methodology (Agile movement)?**

Agile development methodology provides opportunities to assess the direction of a project throughout the development lifecycle. This is achieved through regular cadences of work, known as sprints or iterations, at the end of which teams must present a potentially shippable product increment. By focusing on the repetition of abbreviated work cycles as well as the functional product they yield, agile methodology is described as “iterative” and “incremental.” In waterfall, development teams only have one chance to get each aspect of a project right. In an agile paradigm, every aspect of development — requirements, design, etc. — is continually revisited throughout the lifecycle. When a team stops and re-evaluates the direction of a project every two weeks, there’s always time to steer it in another direction.

The results of this “inspect-and-adapt” approach to development greatly reduce both development costs and time to market. Because teams can develop software at the same time they’re gathering requirements, the phenomenon known as “analysis paralysis” is less likely to impede a team from making progress. And because a team’s work cycle is limited to two weeks, it gives stakeholders recurring opportunities to calibrate releases for success in the real world. Agile development methodology helps companies build the right product. Instead of committing to market a piece of software that hasn’t even been written yet, agile empowers teams to continuously re-plan their release to optimize its value throughout development, allowing them to be as competitive as possible in the marketplace. Development using an agile methodology preserves a product’s critical market relevance and ensures a team’s work doesn’t wind up on a shelf, never released

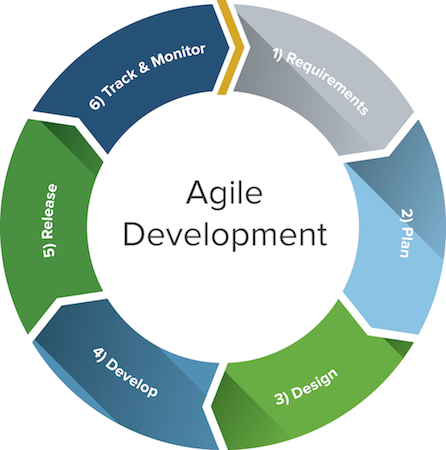


Fig 2.1 Agile development

**2.1.2 Project Effort and Time, Cost Estimation**

Now we will calculate Effort, Cost and Project Duration Estimation using The Basic COCOMO Model which is itself a Heuristic Estimation Technique.

KLOC= lines of code (in thousands) =1

a=3

b=1.12

c=2.5

d=0.35

* **Estimation of Development Effort:**

**Effort = a \* KLOCb person/months**

**= 3(1)1.12 PM**

**= 3 PM**

* **Estimation of Development Time:**

**Tdev = c (Effort)d Months**

= 2.5(3)0.35 Months

= 3(1.4689) Months

= 3.6722 Months

**2.2 PROJECT SCHEDULING**

The project schedule provides a road map for a software project manager. If it has been properly developed, the project schedule defines the task and milestones that must be tracked and controlled as the project proceeds. Tracking can be accomplished in a number of different ways. In agile scrum methodology we estimate time during sprint planning meeting and we track this time using Gantt chart.

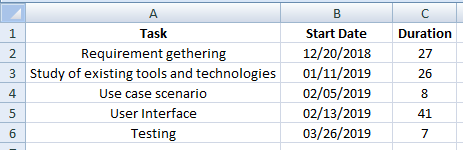
****

Fig: 2.2 Tasks and Duration

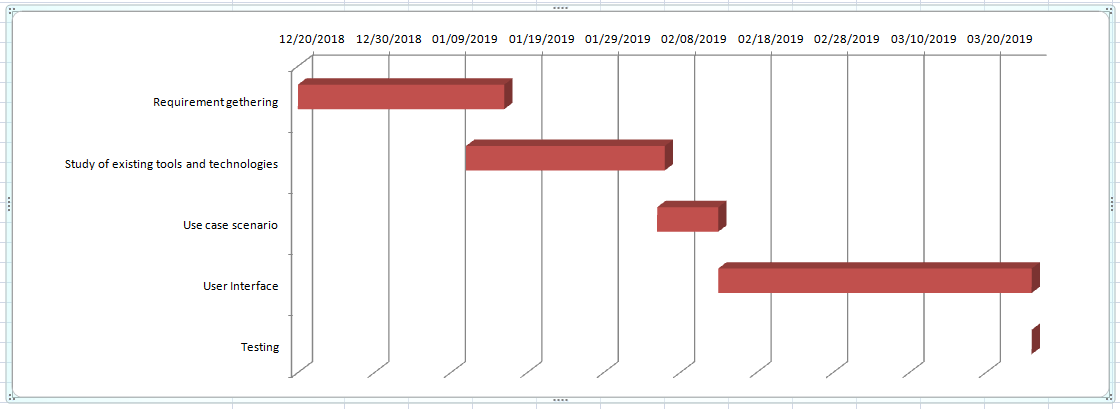


Fig: 2.3 Gantt Chart