

## CHAPTER - 2

### PROJECT MANAGEMENT

#### 2.1 PROJECT PLANNING

---

##### 2.1.1 Concept:

**Project management** is the process and activity of planning, organizing, motivating, and controlling resources, procedures and protocols to achieve specific goals in scientific or daily problems.

Success for project means:

- It must be completed.
- It must be completed within specific budget.
- It must be completed within allocated time.
- The customer must be satisfied.

##### 2.1.2 Project Development Approach and Justification:

There are various models available in software but the processes model used for developing this project is the “Agile Development Model”.

**Agile software development** is a group of software development methods in which requirements and solutions evolve through collaboration between self-organizing, cross-functional teams. It promotes adaptive planning, evolutionary development, early delivery, continuous improvement and encourages rapid and flexible response to change.

The Agile Manifesto, which first laid out the underlying concepts of Agile development, introduced the term in 2001

The Agile Manifesto is based on 12 principles:

1. Customer satisfaction by rapid delivery of useful software
2. Welcome changing requirements, even late in development
3. Working software is delivered frequently (weeks rather than months)
4. Close, daily cooperation between business people and developers
5. Projects are built around motivated individuals, who should be trusted
6. Face-to-face conversation is the best form of communication (co-location)

7. Working software is the principal measure of progress
8. Sustainable development, able to maintain a constant pace
9. Continuous attention to technical excellence and good design
10. Simplicity—the art of maximizing the amount of work not done—is essential
11. Self-organizing teams
12. Regular adaptation to changing circumstances

There are many specific agile development methods. Most promote development, teamwork, collaboration, and process adaptability throughout the life-cycle of the project.

### **Iterative, incremental and evolutionary**

Most agile methods break tasks into small increments with minimal planning and do not directly involve long-term planning. Iterations are short time frames (timeboxes) that typically last from one to four weeks. Each iteration involves a cross-functional team working in all functions: planning, requirements analysis, design, coding, unit testing, and acceptance testing. At the end of the iteration a working product is demonstrated to stakeholders. This minimizes overall risk and allows the project to adapt to changes quickly. An iteration might not add enough functionality to warrant a market release, but the goal is to have an available release (with minimal bugs) at the end of each iteration. Multiple iterations might be required to release a product or new features.

### **Efficient and face-to-face communication**

No matter what development disciplines are required, each agile team will contain a customer representative, e.g. Product Owner in Scrum. This person is appointed by stakeholders to act on their behalf and makes a personal commitment to being available for developers to answer mid-iteration questions. At the end of each iteration, stakeholders and the customer representative review progress and re-evaluate priorities with a view to optimizing the return on investment (ROI) and ensuring alignment with customer needs and company goals.

In agile software development, an information radiator is a (normally large) physical display located prominently in an office, where passers-by can see it. It presents an up-to-date summary of the status of a software project or other product. The name was coined by Alistair Cockburn, and described in his 2002 book *Agile Software Development*. A build light indicator may be used to inform a team about the current status of their project.

### **Very short feedback loop and adaptation cycle**

A common characteristic of agile development are daily status meetings or "stand-ups", e.g. Daily Scrum (Meeting). In a brief session, team members report to each other what they did the previous day, what they intend to do today, and what their roadblocks are.

### **Quality focus**

Specific tools and techniques, such as continuous integration, automated unit testing, pair programming, test-driven development, design patterns, domain-driven design, code refactoring and other techniques are often used to improve quality and enhance project agility.

### 2.1.3 Milestones and Deliverables

**Milestones:**

When planning for the project a series of milestones are established. These milestones are end-point for software activity. At each milestone in our project some formal output for project generated. It may be in form of report.

Some milestones in our project are discussed as below:

- Requirements Gathering and study API
- Requirement of various functionality for GUI
- Creating Procedural Design
- Standard System Flow Diagram
- Designing of GUI Form
- Add functionality Coding and Implementation
- Unit Testing combined with validations, flow of contents
- System Testing

**Deliverables:**

A deliverable is a project report that is delivered to use. It is usually delivered at the end of some major phase such as specification and design.

## 2.2 PROJECT SCHEDULING

In project management, a **schedule** is a listing of a project's milestones, activities, and deliverables, usually with intended start and finish dates. Those items are often estimated in terms of resource allocation, budget and duration, linked by dependencies and scheduled events.

Project scheduling is an activity that distributes estimated efforts across the planned duration by allocating the effort to specific software engineering tasks. Scheduling the project task is an important project planning activity. It involves deciding which tasks would be taken up and when. Based on the planned duration of required tests and collection of resources to complete those tasks projected, completion date is calculated. We have prepared Gantt chart which is shown as below.

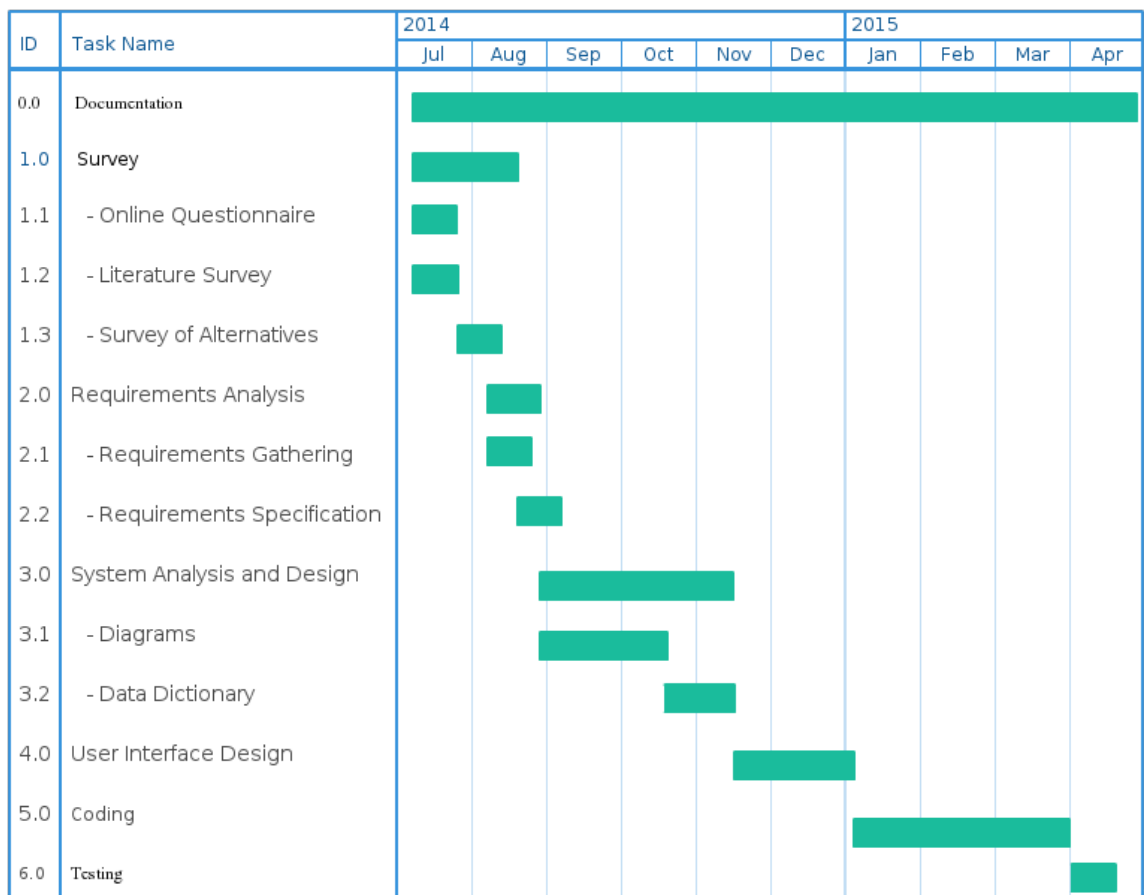


Fig. 2.1 Gantt Chart

## 2.3 RISK MANAGEMENT

---

**Risk management** is the identification, assessment, and prioritization of risks (defined in ISO 31000 as the effect of uncertainty on objectives) followed by coordinated and economical application of resources to minimize, monitor, and control the probability and/or impact of unfortunate events or to maximize the realization of opportunities.

Risks can come from uncertainty in financial markets, threats from project failures (at any phase in design, development, production, or sustainment life-cycles), legal liabilities, credit risk, accidents, natural causes and disasters as well as deliberate attack from an adversary, or events of uncertain or unpredictable root-cause. Several risk management standards have been developed including the Project Management Institute, the National Institute of Standards and Technology, actuarial societies, and ISO standards. Methods, definitions and goals vary widely according to whether the risk management method is in the context of project management, security, engineering, industrial processes, financial portfolios, actuarial assessments, or public health and safety.

### Principles of risk management

The International Organization for Standardization (ISO) identifies the following principles of risk management:

Risk management should:

- create value – resources expended to mitigate risk should be less than the consequence of inaction, the gain should exceed the pain
- be an integral part of organizational processes
- be part of decision making process
- explicitly address uncertainty and assumptions
- be systematic and structured process
- be based on the best available information
- be tailorable
- take human factors into account
- be transparent and inclusive
- be dynamic, iterative and responsive to change
- be capable of continual improvement and enhancement
- be continually or periodically re-assessed

There are some important techniques in this Risk Management:

- Risk identification
- Risk analysis
- Risk planning

### 2.3.1 Risk Identification

It is process of identifying potential risks. Risks are about events that, when triggered, cause problems or benefits. Hence, risk identification can start with the source of our problems and those of our competitors (benefit), or with the problem itself.

- Source analysis - Risk sources may be internal or external to the system that is the target of risk management (use mitigation instead of management since by its own definition risk deals with factors of decision-making that cannot be managed).

Examples of risk sources are: stakeholders of a project, employees of a company or the weather over an airport.

- Problem analysis- Risks are related to identified threats. For example: the threat of losing money, the threat of abuse of confidential information or the threat of human errors, accidents and casualties. The threats may exist with various entities, most important with shareholders, customers and legislative bodies such as the government.

### 2.3.1 Risk Analysis:

- Project Risk: Project risk concern various forms of budgetary, schedule, personal, resource, and customer-related problems, an important project task is schedule slippage.
- Technological Risk: Technical risk concern potential design, implementation, interfacing , testing and maintained problem .most technical risk occur due to insufficient knowledge about the product .in other words most technical risk occur because a problem is found to be difficult to solve than previously thought.
- Business Risk: Business risk include risk of building an excellent product that no one wants, losing budgetary or personal commitments , etc.

### 2.3.2 Risk Planning:

Here is how we deal with all the above said risks:

- Project Risk: We have gathered the required resources for project and designed the schedule which very flexible as we are following agile software development model.
- Technological Risk: First of all we learnt the Android Technology well in order to avoid any kind of misunderstanding and misinterpretation in terms of this technology.

-

- Business Risk: We asked several questions to users to get their opinions about the how they manage time and become productive.

## **2.4 ESTIMATION**

---

### **2.4.1 Effort Estimation:**

- Effort estimation was done by keeping in the mind importance of all the two modules of the whole project.
- The major part of this project was training and leave module.
- The next important thing is time attendance management of the Users.
- In a way, system provides various functions for analysis the User's performance in the organization related to their work.

### **2.4.2 Cost Analysis:**

The Business model followed here to develop the application aims at cost effective budget. The targeted application aims at the common man who neither is techno savvy nor will be interested to buy expensive applications. The cost effectiveness of the application was the important factor which had to take care of throughout the application development. The application uses some of the best Open Source resources currently used in this era for development. These not only cuts down the cost but also helps in being portable.