Chapter 4
Studies of Project Tasks

ABSTRACT

Among the projects available in sourceforge.net, the three top ranked projects are selected for studying the pattern of project tasks over a period of 6 years (2005-2011). It is found that the number of tasks in projects decrease with time in these projects. It is also observed that the amount of time taken to complete the task decrease with time. The developers alloted to tasks in a project, success rate the developers complete the tasks completely, and active contribution to the project by completing the alloted tasks of the volunteers are also studied.

INTRODUCTION

Software products are examples of complex systems. The complexity of software is due to the large number of requirements such a product should satisfy. Additionally, the development of software demands diverse skill sets, technically and other, which one person cannot possibly posses. Any software which is expected to run in real world is therefore built by a group of developers. This feature of software development has forced the practitioners to adopt some design principles to accommodate multiple developers working on a project.

DOI: 10.4018/978-1-5225-3707-6.ch004

The earliest attempt to solve this problem wasto divide the software development process into various related activities. All the process models and methodologies used in software engineering embody this idea. The basic activities of analysis, design, code, test and maintenance are common to all devel- opment methods. This separation of concern gives an opportunity to divide the software development into tasks which can be allotted to each individual developer or groups ofdevelopers.

Another way to solve the problem of synchronising multiple developers isfollowing the design principle of modularity. Modularity helps to isolate functional elements of the system. One module may be debugged, improved, or extended with minimal interaction to system discontinuity. As important as modularity is specification.

The key to production success of any modular construct is a rigid specification of the interfaces. They also help in the maintenance task by supplying the documentation necessary to train, understand, and provide maintenance.

The principles of separation of concern and modularity are applied extensively in development of FOSS. Modularising the software component allows parallelism and thus speeds up the process of development. Modularising also allows the developers to select a particular task to complete. In this chapter, the studies of such project tasks are undertaken. The number of tasks in each project, time taken for completing the tasks, the allocation of tasks to developers and the amount of tasks they complete are studied.

Number of Project Tasks

The activity in the FOSS project can be measured in different ways. The most definite metric is the number of commits made to the source code. But that does not capture the various other tasks undertaken by participants in the project. There are very broad set of activities that occur in a FOSS project like requesting new features, reporting bugs, support requests, documentation, localisation and internationalisation. To cover all the possible tasks that occur in a project, the Table PROJECT_TASK is used. The structure of this table is given in Table 1.

The procedure for finding the number of tasks in the project P is as follows

```
FOR i in (1 to 54)
DO
FOR ALL in Project_Task
```