DSPPTOOL: A TOOL TO SUPPORT DISTRIBUTED SOFTWARE PROJECT PLANNING

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS
OF
THE MIDDLE EAST TECHNICAL UNIVERSITY

BY

SEVİL YILMAZ YAĞIZ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE
OF MASTER OF SCIENCE
IN
THE DEPARTMENT OF INFORMATION SYSTEMS

JANUARY 2004

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

_____

Approval of the Graduate School of Informatics

_____

Prof. Dr. Neşe YALABIK

Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

_____

Assoc. Prof. Onur DEMİRÖRS

Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

_____

Assoc. Prof. Onur DEMİRÖRS

Supervisor

Examining Committee Members

Prof. Dr. Semih BİLGEN _____

Assoc. Prof. Dr. Nazife BAYKAL _____

Assoc. Prof. Dr. Onur DEMİRÖRS _____

Assoc. Prof. Dr. Ali DOĞRU _____

Dr. Altan KOÇYİĞİT _____

ABSTRACT

DSPPTOOL: A TOOL TO SUPPORT DISTRIBUTED SOFTWARE PROJECT PLANNING

Yağız, Sevil Yılmaz

M.S., Department of Information Systems
Supervisor: Assoc. Prof. Dr. Onur Demirörs

January 2004, 68 pages

This thesis focuses on the development of a distributed software project planning tool that enables more than one participant to prepare the different parts of the project scope, schedule and task assignment by allowing to utilize the predefined organizational level processes. For this purpose, we discuss the need for a distributed software project planning tool, identify tool requirements and compare available tools with respect to the requirements. In addition, we evaluate the tool based on two criteria: first one is the tool's adequacy to meet the identified functional attributes and the second one is the validation of the tool by utilizing the data of the project schedule of a real project. This tool enables preparation of project scope, schedule and task assignments in a more effective, accurate and seamless way.

Keywords: Project Planning, Distributed Planning, Task Assignment

ÖZ

DYPPARAÇ: DAĞITIK ORTAMDA YAZILIM PROJE PLANLAMASINI DESTEKLEYEN BİR ARAÇ

Yağız, Sevil Yılmaz

Yüksek Lisans, Bilişim Sistemleri Bölümü
Tez Yöneticisi: Doç. Dr. Onur Demirörs

Ocak 2004, 68 pages

Bu tez dağıtık ortamda yazılım proje planlamasını destekleyen bir aracın geliştirilmesini hedeflemektedir. Bu araç proje kapsamının, zaman planının ve iş atamalarının, organizasyonel seviyedeki süreçleri kullanalarak birden fazla katılımcı tarafından hazırlanabilmesini sağlar. Bu amaçla dağıtık ortamda yazılım proje planlamasının gereksinimleri tartışılmış, geliştirilen aracın gereksinimleri tanımlanmış ve varolan yazılım araçları karşılaştırılmıştır. Ayrıca geliştirilen araç iki kriter temel alınarak değerlendirilmiştir: İlk kiriter, aracın belirlenmiş fonksiyonel nitelikleri karşılayıp karşılamadığının belirlenmesi, ikinci kriter ise gerçek bir projenin proje plan verileri kullanılarak aracın kullanılırlığının tasdik edilmesidir. Bu araç proje kapsamının, zaman planının ve iş atamalarının daha etkin, doğru ve sorunsuz bir şekilde hazırlanmasını sağlamaktadır.

Anahtar Kelimeler: Proje Planlama, Dağıtık Planlama, İş Atama

To my precious and unique husband,
Engin YAĞIZ

ACKNOWLEDGEMENTS

I first thank my advisor Dr. Onur Demirörs for providing guidance, encouragement and patience, which promoted this study. I am grateful to Dr. Onur Demirörs for his inspiration and desire.

I am forever indebted to my husband, Engin, for his support, endless patience and precious comments about my study.

Finally, words cannot truly express my deepest gratitude to my mother, Cevriye, father, Cevdet, sister, Selin and brother, Tolga who always gave me their love and emotional support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS AND ACRONYMS

ADM:	Arrow Diagramming Method
ATC:	Air Traffic Control System
CDM:	Conditional Diagramming Method
DAO:	Database Access Objects
DSPMtool:	Distributed Software Project Management Tool
DSPPTool:	Distributed Software Project Planning Tool
GERT:	Graphical Evaluation and Review Technique
GUI:	Graphical User Interfaces
HR:	Human Resources
JDBC:	Java Database Connectivity
JRE:	Java Runtime Environment
MIS:	Management Information System
MS:	Microsoft
OSS:	Open Source Software
PDM:	Preceding Diagramming Method
PERT:	Program Evaluation and Review Technique
PMBOK:	Project Management Body Of Knowledge
RMI:	Remote Method Invocation
UML:	Unified Modeling Language
WBS:	Work Breakdown Structure

CHAPTER 1

INTRODUCTION

1.1 Background and Rationale of the Study

The Standish Group reveals that Corporate America spends more than $275 billion at each year on approximately 200,000 application software development projects [1]. Also Donna Dufner [2] stated that one of the largest problems facing management today is the management of software development. It is estimated that over 1/3 of all software development projects, upwards of 80 billion dollars for each year, are abandoned before completion [2]. Only 18% of completed software projects are within budget or on time [3].

As stated above, there exist many problems in software project management. We believe that one of the reasons of these failures in software projects is related with the difficulties of software project planning.

One of the problems of software project planning is improper project scope definition. In some cases, boundary of the project scope may not be defined correctly. This improperly defined boundary may change throughout the project life cycle [4]. This change causes the waste of the effort spent for the parts extracted in the new scope. Because of being useless of their previous works, the project members becomes frustrated and unmotivated.

Another problem is related with the task assigment process. During software project planning, in most cases, tasks are assigned to the project members directly by the project manager [5]. Genarally, project managers do not know details of the tasks. Therefore, assigned tasks may not be in sufficient detail and/or may not

be completed in the specified time. Since the task assignments are prepared without taking the opinion of the developers, they shift responsibilty of incompleted tasks to managers.

For a project to success, a well-defined project plan is essential [2]. To say that a project plan is well defined, the most important parts of the plan, which are the project scope, schedule and task assignments, should be well defined. To prepare such a project plan, firstly the boundary of the project scope should be defined concretely. Then project scope should be decomposed in to the project tasks. To make the project plan well defined, realistic and achievable, the project members, which know the details of the work, should participate in preparation of the plan.

Nowadays, the number of software projects developed in a distributed environment increases [6]. In such a situation, to provide the contribution of the project members to project planning, software project planning activities could be performed in a distributed environment [7][8].

Another necessity is to utilize the organizational processes while preparing the project plan. For software project planning, most of the development activities may be similiar for different projects. These activities may be defined at the organizational level to be used later in the project planning.

As a result, preparing a well-defined project plan is a large and complex process. It is not an isolated process from the project members [1], it is most often the result of a collective effort [9]. It begins before technical work starts, continues as the development evolves and ends when the software delivered [10]. To provide the individuals' contribution [11], it is required to include them into project planning process. In this study, we focused on developing a distributed project planning tool. This tool enables more than one people prepare the different parts of the project scope, schedule and task assignments by utilizing the organizational processes.

1.2 Problem Statement and the Need to Develop an Automated Support for Project Planning

Preparing project scope, time and resource and then plans of a project is the main part of the project planning. It is a complex process, which includes the identification of the project activities, project members and roles, making the personnel-role mapping, definition of the teams in the project, making the assignments of the activities to the project members and/or teams and setting start and finish time of the assignments.

To make all these manually is overwhelming and time consuming. Existing project management tools lighten the load of the project managers. However, these tools do not enable to collaboration of people to prepare the different parts of the plan. In a result, generally people to whom activities are assigned do not involve in the planning process. Besides, whenever you want to prepare project scope, schedule and task assignments for a project in an organization, each time, you have to think and write down the activities again and again. That is, utilization of organizational processes in project planning is not supported. In addition, while realization of the assigned task, finding the project documents to be used and the documents to be produced to complete the assigned task is difficult and time consuming. All these difficulties contributes to the delay of the assigned tasks. To summarize, convetional project management tools do not provide the following utilities altogether:

- Defining the activities in a hirerchical structure based on the organizational processes to give the opportunity of utilizing the organizational processes while preparing the project plans.
- Defining the roles at the organizational level to be used in personnel-role mapping for a project.
- Construction of teams for a project in a hirerchical structure to provide assignment of activities to the project members and/or teams.
- Commitment of assignments by the individuals to whom activities are assigned to provide the contribution of the project members.

- Providing the project individuals with the To Do List including the tasks assigned to those individuals (Project Assginment To Do List).
- Providing the team leaders with the To Do List including the issues to be done while preparing the project schedule (Project Schedule To Do List).
- Effective sharing of project documents through the life cycle of the project to facilitate project activities.

To bring a conclusion, these tools do not facilitate sharing of the project information effectively among the project members. Moreover, they do not enable the reflection of organizational level information to the project planning activities.

1.3 The Approach

The following statement is set as the aim of this thesis:

- To develop a distributed project planning tool that enables more than one people prepare the different parts of the project scope, schedule and task assignment by allowing the use of the predefined organizational level activities and enabling the effective use of the project information through the project life cycle.

Distributed project planning tool can improve the completion of the assigned tasks on time, because all individuals will be aware of their assignments at the project planning phase. Also the tool enables the indivduals, who are located in geographically different locations, participate in creation of the project scope and schedule, and reduces the communication problems. It prevents the re-entrance/re-creation of project information such as roles, activities used while project schedule is prepared for each project. Also the tool enables the definition of organizational level information, such as personnel, roles and activities and it provides the project managers and/or the individuals, who participate in preparing project scope and schedule, with the use of the organizational level information in an effective way. Moreover, it provides the effective use of the project related information by aspect of easy access of project documents through the life cycle of the project.

This aim is established to remove the problems that discourage managers and engineers from making project planning cooperatively, to enable the reflection of the predefined organizational level information to the project plans and to make the effective use of the project information through the project life cycle.

To fulfill the aim of this thesis, the following objectives are established:

- To provide a tool to support distributed software project planning
- To enable the use of the predefined organizational level information while preparing the project schedule
- To enable the effective use of the project documents through the project life cycle

The usage scenario of the tool can provide necessary information to explain our approach. Firstly the project manager enters the general project information to make the definition of the project and then he/she identifies the project members with their roles and creates the project teams by selecting the team members from the project members. After that, he/she starts preparing the project scope and then the schedule by selecting the predefined activies and assigning the predefined activities to the members or teams. Each team leader continues the project schedule preparation by detailing the Work Breakdown Structure (WBS) items assigned to them and making the detailed activity assignments to the team members. Each project member will be able to commit the assignments made to them. At the end of the project planning, project manager approves the prepared project schedule.

Moreover which project documents are to be used and which ones are to be produced while performing the assigned tasks will also be defined in the planning process. Access rights of the project members to the project documents will also be defined by means of the tool. And all individuals may see their task assignments (Project Assignment To Do list) when they log into the system.

Also this tool has a capability of exporting the prepared project schedule as a text file. This text file can be imported into MS Project 2000. By this way, the integration of our tool with the MS Project is provided. By means of this integration, project schedule data (activities, start and finish dates, activitiy

dependencies, assigned people and their percentage) created in our tool is transfered to the MS Project and then the different views of the project schedule may be visualized.

In the requirements development and design phases, we used UML methodology. In the requirements phase, we prepared use case diagrams and detailed the use cases by flow of events. In the design phase, we prepared class diagrams and we designed database tables. In the implementation phase, we developed the tool by using Java SDK 3.1 and JBuilder 6.0 java development environment. As known, java provides platform independent architecture, which enables widespread use of the tool. Moreover, to enable the tool run in a distributed environment, RMI technology were used. The underlying database management system is Microsoft Access, which seems to limit the application to MS Windows environment. However, the layered architecture of the tool separates the database layer from bussiness logic and interface layers by using Database Access Objects (DAO) patterns, therefore providing an effective way to port the system to other environments.

The tool is validated by utilizing the project schedule provided by MilSOFT A.Ş.. The company gave a project schedule of one of their completed projects with the provision of hiding the project's and project members' names.

For the sake of understandablity, I give the name, ***MilProject***, to the project whose project schedule is given by MilSOFT. In the rest of the thesis, ***MilProject*** will be used to call this project. The results are discussed in the Chapter 5. Also the screen shot of the project schedule information of the ***MilProject*** project is provided in the same chapter

1.4 Thesis Structure

Chapter 2 provides the related research including software project management concepts, automation support to software project planning process. In addition, reviews of the analyzed project planning support tools are included.

Chapter 3 presents the requirement specifications of the tool developed, which is called DSPPTool. The requirements are elicited by using use case analysis method.

Chapter 4 provides the design of the tool including module decomposition, data decomposition and interface description. In the interface description, graphical user interfaces and interface with the MS Project 2000 are explained.

Chapter 5 includes an evaluation of the developed tool.

Chapter 6 provides a conclusion to the study and includes the future work directions for automated support to software project planning process.

CHAPTER 2

RELATED RESEARCH

In our study, we take Project Scope Management, Project Time Management and Project Human Resource Management Knowledge Areas as a basis. The detailed information about these knowledge areas and the processes of these knowledge areas are given in the first three sections, and then the analysis of the existing project planning tools are presented in the fourth section.

2.1 Project Scope Management

In the Project Scope Management Knowledge Area, initiation, scope planning, scope definition, scope verification and scope change control processes are grouped. These processes consist of the prerequisite activities for the preparation of the WBS and the preparation of high level WBS. To make the project plan, firstly scope should be defined. Project scope is the fundamental stones of the project plan. Because if the scope is not described clearly and explicitly, the boundary of the system to be developed cannot be drawn correctly. And all the individuals in the project may go to the different directions. The work to be done to complete the project may deviates from the actual required work. Then this situation makes the individuals unmotivated [12]. And the successful completion of the project goes in danger. Therefore, these processes identified in this project knowledge area are the fundamental of project scope preparation.

2.1.1 Initiation

In the PMBOK, Initiation is described as the process of formally authorizing a new project or that an existing project should continue into its next phase. The input of this process are product description, which describes the characteristics of the product or services to be developed as a result of the project; strategic plan, which includes the organization's strategic goals; project selection criteria and historical information about the results of the previous project selection decision and previous project performance. The expert assesses these inputs by use of decision models and calculation methods. Finally the following outputs are developed: project charter, constraints and assumptions. Project charter is a document, which formally authorizes the project. It includes the business need and the product description. At the end of this process, project manager is identified and assigned to the project.

2.1.2 Scope Planning

In the scope planning, project scope is made detailed progressively. The inputs are project description, project charter, constrains and assumptions. Product is analyzed to make the better understanding of it [12]. Benefit/cost analysis is performed by estimating the tangible and intangible costs and benefits of various project and product alternatives and then the identified alternatives are assessed by using financial measures, such as return on investment or payback period [12]. To generate the different approaches to the project alternative identification techniques, such as brainstorming, are used [12]. The outputs of scope planning are the scope statement and scope management plan with the supporting detail. The scope statement establishes a basis for the future project decisions and for the common understanding among the stakeholders [12]. In the scope management plan, change mechanism of project scope is identified. It includes how the changes to the project scope are integrated to the project [12].

2.1.3 Scope Definition

In the PMBOK, it is stated that to improve the accuracy of estimates (cost, resource and duration), to establish a performance measurement and control baselines and to make responsibility assignment clear, major project deliverables should be subdivided into more manageable parts. The inputs of the scope definition are scope statement, constraints, assumptions and historical information. By performing scope definition, a WBS from a previous project can be used as a template [12]. By decomposing in the scope definition [12]:

(1) Firstly major project deliverables of the project, including project management are identified.

(2) Decision is made whether adequate cost and duration estimates can be developed at this level of detail for each deliverable. For each deliverable, if the decision is yes go to the step 4, otherwise go to the step 3.

(3) Sub-components of deliverables are identified. While finding sub-components, how the work of project will be organized and the work of project accomplished should be considered.

(4) Correctness of the decomposition is verified.

- Are the lower-level items both necessary and sufficient for completion of the decomposed item? If not, the sub-components must be modified.
- Is each item clearly and completely defined? If not, the descriptions must be revised.
- Can each item be appropriately scheduled? Budgeted? Assigned to a specific organizational unit (e.g. team or person) who will accept the responsibility for satisfactory completion of the item? If not, revisions are needed to provide adequate management control.

The outputs from the scope definition are WBS and the updates to the scope statement.

2.1.3.1 Work Breakdown Structure (WBS)

A Work Breakdown Structure (WBS) is defined as a method of representing, in a hierarchical manner, the parts of a process or product [13]. It is considered one of

the best project management tools in existence today. Work breakdown structures are categorized into three groups as the following [13].

1. *Process WBS:* The process WBS is used by project managers to manage their software engineering projects. The top level of the WBS identifies the project by name. The second level identifies the major work elements to be done, such as planning, organizing, requirements analysis, design, coding, testing, and so forth. The third level describes the more detailed activities that must be completed in order to accomplish the second level activities. The lowest level process elements in a WBS represent tasks. A task is the smallest unit of work that is suitable for assignment to one or two persons for duration of one or two weeks. Figure 1 provides an example of a process WBS.

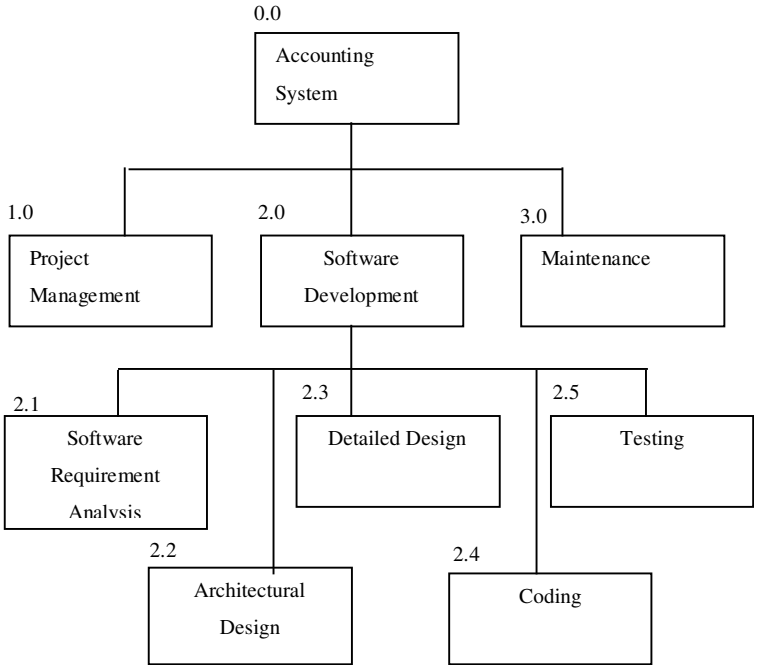

Figure 1 Accounting Work Breakdown Structure (Process)

2. *Product WBS:* A product WBS illustrates the components and interfaces of the product to be developed or produced and relates the elements of the product to each other and to the end product. It specifies the hardware, software and data

that together completely define a project deliverable. The top level of the product WBS identifies the product by name. Other elements of a product WBS are discrete, identifiable items of hardware, software and data. Figure 2 gives an example of product WBS.

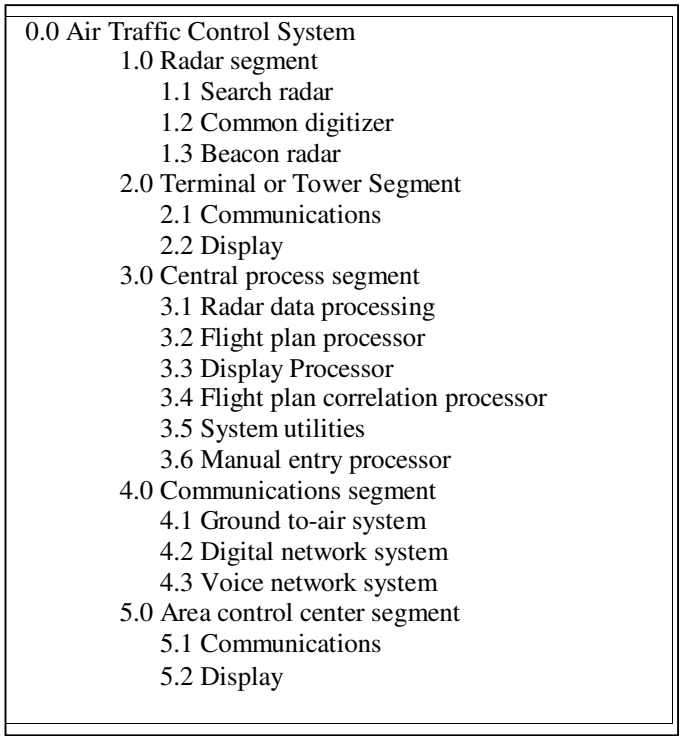

Figure 2 ATC Work Breakdown Structure (Product)

3. *Hybrid WBS:* A hybrid WBS is a kind of WBS beginning with process elements and interleaving product with process elements or beginning with product elements and interleaving processes. The rationale behind this approach is that processes produce products. Sub products, in turn, require development processes. The terminal elements are product components. (See Figure 3)

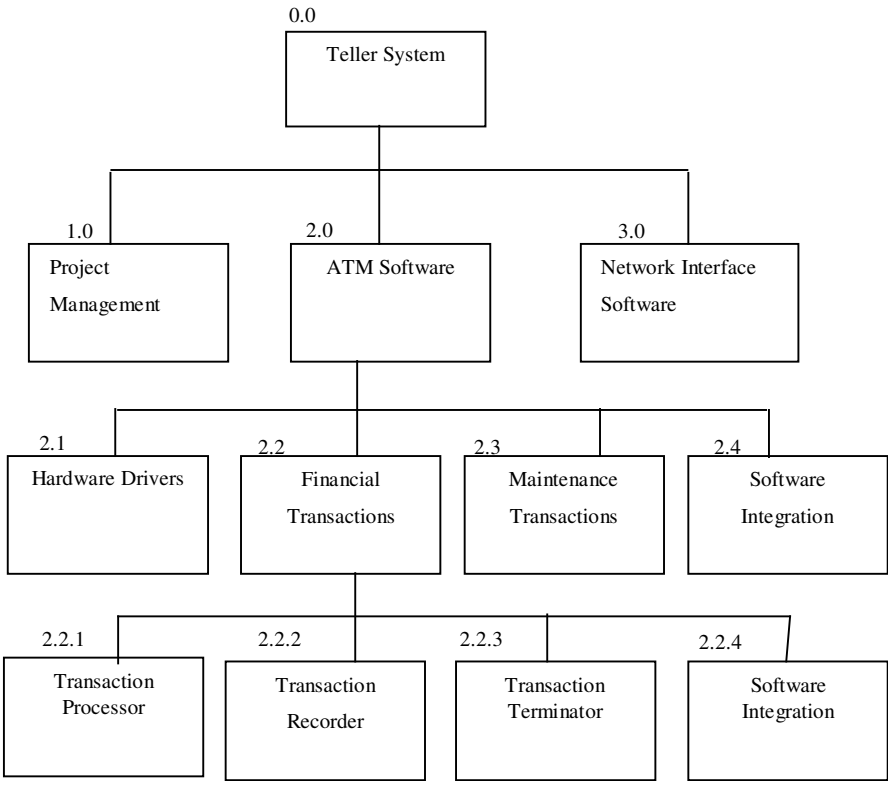

Figure 3 Teller System Work Breakdown Structure (Hybrid)

In their paper, Fairly and Thayer describe the steps to develop the WBS as follows [13]:

1. Determine the purpose of the WBS. For example, is it to identify the elements of a product? To assign work to individuals? Etc.
2. Identify the top of the WBS.
3. Partition the WBS into its major components.
4. Partition each of the major components into 7± 2 elements.
5. Terminate the portioning when the goal of WBS is reached. For purposes of assigning work to individuals, the goal may be to decompose the WBS until each lowest level task is a work element for one to two people for one to two weeks.
6. Develop a WBS dictionary entry for each lowest-level element of the WBS.
7. Document the result.

As seen, defined steps to develop WBS are same as the steps of decomposition given in the scope definition. In the PMBOK, it is stated that as with the scope statement, the WBS is used to develop a common understanding of project scope. Each descending level represents increasingly detailed descriptions of the project deliverables [12].

2.1.4 Scope Verification

Scope verification is described in the PMBOK as the process of obtaining formal acceptance of the project scope by the stakeholders. It requires reviewing deliverables and work results to ensure that all were completed correctly and satisfactorily [12]. The inputs are work results - which deliverables have been fully or partially completed, - output of project plan execution, product documentation produced to describe the project's products, WBS providing the scope definition in detail, scope statement and project plan. In the scope verification, inspection techniques are performed, such as reviews, product reviews, audits and walkthroughs, to determine whether results conform to requirements [12]. The output is formal acceptance which shows that client or sponsor has accepted the product of the project phase or major deliverables.

2.1.5 Scope Change Control

Scope change control is concerned with [12]:

- Influencing the factors that create scope changes to ensure that changes are agreed upon
- Determining that a scope change has occurred
- Managing the actual changes when and if they occur.

The inputs are WBS defining project's scope baseline, performance reports providing information on scope performance, such as which interim deliverables have been completed and which have not, change requests to the project scope and scope management plan.

Scope change control defines the procedures by which the project scope may be changed. It includes the paperwork, tracking systems, and approval levels

necessary for authorizing changes [12]. The magnitude of any variations that do occur is assessed with the help of performance measurement technique [12]. The outputs are scope changes, any modification to the agreed upon project scope as defined by the approved WBS, corrective action, anything done to bring expected future project performance in line with the project plan, lessons learned and adjusted baseline.

2.2 Project Time Management

In the Project Time Management Knowledge Area, activity definition, activity sequencing, activity duration estimating, schedule development and schedule control processes are grouped [12]. These processes include the activities to make the WBS more detailed and relating the schedule with the WBS by time aspects of the items in the WBS.

2.2.1 Activity Definition

It is stated in the PMBOK that activity definition involves identifying and documenting the specific activities that must be performed to produce the deliverables and sub deliverables identified in the WBS. The inputs are WBS, scope statement, historical information, constraints, assumptions and expert judgment. In the activity definition, project work packages are decomposed into more smaller, more manageable components to provide better management control by using same technique described in the Scope Definition section [12]. But the difference in here is that the final outputs here are described as activities rather than as deliverables [12]. Activity list from a previous project is used as a template for a new project [12]. The outputs are activity list that contains all activities that will be performed on the project and WBS updates. The WBS and the activity list are usually developed in sequentially, with the WBS being the basis for the development of the final activity list [12].

2.2.2 Activity Sequencing

Activity sequencing is described as in the PMBOK, identifying and documenting interactivity logical relationships. These relationships are established accurately to support the later development of a realistic and achievable schedule [12]. The inputs are activity list, product description, mandatory dependencies, discretionary dependencies, external dependencies and milestones. In the PMBOK, mandatory dependencies are described as those, which are inherent in the nature of the work being done. Discretionary dependencies are described as those, which are defined by the project management team. And External dependencies are described as those, which involve a relationship between project activities and non-project activities. In the activity sequencing, the following techniques may be used [12].

- *Preceding Diagramming Method (PDM):* This is a project networking diagramming method that uses boxes or rectangles (nodes) to represent the activities and connect them with arrows that show the dependencies. It includes four types of dependencies or precedence relationships:
    1. Finish-to-start: the initiation of the work of the successor depends upon the completion of the work of the predecessor.
    2. Finish-to-finish: the completion of the work of the successor depends upon the completion of the work of the predecessor.
    3. Start-to-start: the initiation of the work of the successor depends upon the initiation of the work of the predecessor.
    4. Start-to-finish: the completion of the work of the successor depends upon the initiation of the work of the predecessor.
- *Arrow Diagramming Method (ADM):* This is a project networking diagramming method that uses arrows to represent the activities and connects them at nodes to show their dependencies. ADM uses only finish-to-start dependencies.
- *Conditional Diagramming Method (CDM):* This method allows for non-sequential activities such as loops or conditional branches.

The outputs are activity list updates and project network diagrams which are schematic displays of the project's activities and the logical relationships among them.

2.2.3 Activity Duration Estimating

In the PMBOK, activity duration estimating is defined as a process of taking information on project scope and resources and then developing duration for inputs to schedules. The inputs are activity list, constraints, assumptions, resource requirements, which affects the duration of most activities, resource capabilities, which are the capabilities of the human and material resources assigned to the activities, historical information and identified risks, which can have a significant influence on duration. The following techniques may be used in activity duration estimating: [12]

- *Expert judgment:* Estimation is difficult in its nature. Therefore expert judgment guided by historical information should be used.
- Analogous estimating: Analogous estimating, also called top-down estimating, means using the actual duration of a previous, similar activity as a basis for estimating the duration of the future activity.
- Reserve time (contingency): Project team may choose to incorporate an additional time frame, called time reserve, contingency or buffer, that can be added to the activity duration.

The outputs are activity duration estimates, basis of estimates and activity list updates.

2.2.4 Schedule Development

In the PMBOK, schedule development is described as determining start and finish dates for project activities. Project schedule development process must often be iterated prior to determination of the project schedule [12]. The inputs are project networks diagrams, activity duration estimates, resource requirements, resource pool description, calendars, constraints, assumptions, leads and lags, risk management plan and activity attributes. Resource pool description is described in

PMBOK as, knowledge of what resources will be available at what time and in what patterns. Moreover lag is described as putting a delay (lag) between two activities in the schedule. Leg is described as for example in a finish-to-start dependency, the successor activity starts the specified duration (leg) before the predecessor has completed. The following techniques may be used in schedule development: [12]

- *Mathematical Analysis:* involves calculating theoretical early and late start and finish dates for all project activities. The examples are Critical Path Method, Graphical Evaluation and Review Technique (GERT) and Program Evaluation and Review Technique (PERT)
- *Duration Compression:* looks for ways to shorten the project schedule without changing the project scope.
- *Project Management Software:* is used to assist with schedule development. These products automate the calculation of the mathematical analysis and resource leveling and thus allow for rapid consideration of many schedule alternatives.

The outputs are project schedule including at least planned start and expected finish dates for each activity, schedule management plan which defines how changes to the schedule will be managed and resource requirement updates.

2.2.5 Schedule Control

Schedule change control is concerned with [12]:

- Influencing the factors that create schedule changes to ensure that changes are agreed upon
- Determining that the schedule change has occurred
- Managing the actual changes when and as they occur.

The inputs are project schedule, performance reports providing information on schedule performance, such as which planned dates have been met and which have not, change requests to the project schedule and schedule management plan. Schedule change control defines the procedures by which the project schedule may be changed. It includes the paperwork, tracking systems, and approval levels

necessary for authorizing changes [12]. The magnitude of any variations that do occur is assessed with the help of performance measurement technique [12]. Project management software is used to track planned dates versus actual dates and to forecast the effects of schedule changes. The outputs are schedule update, corrective actions and lessons learned.

2.3 Project Human Resource Management

In the Project Human Resource Management Knowledge Area, organizational planning, staff acquisition and team development processes are grouped. These processes include the activities to make the most effective use of the people involved with the project [12].

2.3.1 Organizational Planning

It is stated that in the PMBOK, organizational planning involves identifying, documenting, assigning project roles, responsibilities, and reporting relationships. The inputs are project interfaces, staffing requirements and constraints. The project interfaces are categorized as organizational interfaces among different organizational units, technical interfaces among different technical disciplines and interpersonal interfaces among different individuals working on the project [12]. Staffing requirements define what kinds of competencies are required from what kinds of individuals or groups and in what time frames. Using the role and responsibility definitions of a similar project can help expedite the organizational planning [12]. Organizational procedures can help the project management team with various organizational planning [12]. The outputs are role and responsibility assignments, staffing management plan, organization chart and supporting detail.

2.3.2 Staff Acquisition

In PMBOK, it is stated that Staff acquisition involves getting the needed human resources assigned to and working on the project. The inputs are staffing management plan, staffing pool description and recruitment practices. In the staffing pool description, previous experience, personnel interests, personal

characteristics, availability and competencies and proficiency constraints should be included [12]. In the staff acquisition, it is stated that staff assignments should be negotiated on most project [12]. In some cases, staff may be pre-assigned to the project [12]. And Project procurement management can be used to obtain the human resources to perform project activities. The outputs are assigned project staff and project team directory.

2.3.3 Team Development

Team development includes both [10]:

- Enhancing the ability of stakeholders to contribute as individuals
- Enhancing the ability of the team to function as a teams

The inputs are project staff, project plan, staffing management plan, performance reports and external feedback. The following techniques are used in the team development process [12]:

- *Team-building activities* which include management and individual actions taken specifically and primarily to improve team performance.
- *General management skills*
- *Reward and recognition systems*
- *Collocation* which involves placing all, or almost all, of the most active project team members in the same physical location to enhance their ability to perform as a team.
- *Training*

The outputs are performance improvements in individual skills and/or team behavior and input to performance appraisals.

2.4 Comparison of Tools

There are a number of project planning tools available for software project managers. We examine the following tools to identify their features from operational and functional perspectives. We select eProejct, WebProject, SourceForge and DSPMtool, because they support distributed environment. In

addition, MS Project 2000 is selected because of its widespread use. A brief overview of the tools we examined is given below:

- eProject : This tool was developed by eProject Inc. company in 2000[I]. It is a web-based project. It provides task scheduling and management, team and resource management, time tracking, and reporting facilities for project management.
- Webproject: This tool was developed by Novient company in 2001[II]. WebProject is a three-tiered (database server, application server, client), multiuser, multi-project application written completely in Java. There are no plug-in requirements or other local configurations other than a Java VM and Java enabled browsers.
- SourceForge: This tool was developed by SourceForge company in 2001[III]. The SourceForge source code is itself Open Source Software (OSS) complient. It is a web-service which provides free hosting for OSS project management. It is web-based and developed with PHP and MYSQL in Linux environment. It supports multiple users that can participate in zero or more projects concurrently with other logged on users.
- MS Project 2000 (Microsoft Project Central): This was developed by Microsoft in 2000[IV]. It is flexible project management tool that you can use to control simple or complex projects. It is a single-user tool and runs in Windows environment. As you build a project plan, Microsoft Project calculates and creates a working schedule based on information you provide about the tasks to be done, the people who work on them, the equipment and supplies used to accomplish them, and the costs involved. With Microsoft Project Central, the collobaration of workgroup members is provided. Each work group member can view tasks for all of their projects and can create new

[I] http://www.eproject.com

[II] http://www.wproj.com

[III] http://sourceforge.net

[IV] http://www.microsoft.com/products/info/product.asp

tasks and send them to the project managers for incorporation into the project document.

- DSPMtool: This tool was developed by Hai Eric Lam and Piyush Maheshwari in the university of New South Wales [14]. It presents a single interface which collaborates the different project management tools. This provides gathering, analyzing, integrating and disseminating the outputs of project management process. DSPMtool is a multi-tier and client server application.

To evaluate the project planning tools, we form two checklists based on [15] that interrogate operational and functional aspects. Operational characteristics are related with the installation and operation of tool [15]. Functional characteristics are related with the required functionality of the tool [15]. Trial/demo versions of eProject, WebProject and MS Project 2000 tools are installed and examined. Product documentation of SourceForge, help documentation for Microsoft Project Central and paper about DSPMtool are examined. Moreover documentation of each tool is also used in the assessment.

1. Operational Aspects
   1.1. Does the installation document exist?
   1.2. Is the tool easy to install? (Namely, at what percentage installation is automatic)
   1.3. Does the user manual / help exist?
   1.4. Is the coverage of user manual/help enough to learn and understand the system?
   1.5. Does the tool support commonly used operating systems?
   1.6. Is the execution speed is acceptable? (Response time < 5 seconds)
2. Functional Aspects
   2.1. Does the tool provide the ability to prepare the project schedule with the collaboration of more than one people?
   2.2. If 2.1 is Yes, does the tool enable to inform the individuals, who participate in preparation of the project schedule, about the issues to be done to prepare the project schedule? (By to do list, by email notification, or by sending a message etc.)

2.3. Does the tool provide the interface to define the organizational level information (personnel information, roles and activities)?

2.4. Does the tool enable to define the activities in a hierarchical structure?

2.5. Does the tool enable to utilize the organizational level information while preparing the project schedule?

2.6. Does the tool enable to define the teams in a hierarchical structure?

2.7. Does the tool enable to assign one activity to more than one person or to a team?

2.8. Does the tool enable to assign more than one activity to a person or to a team?

2.9. Does the tool provide the interface to commit the assigned activity?

2.10. Does the tool enable to define the dependencies of activity (pre-activity, post activity)?

2.11. Does the tool enable the project manager to approve the prepared project schedule? (Is there an approval mechanism?)

2.12. Does the tool provide any notification mechanism to inform the individuals about their assigned tasks? (By to do list, by email notification, or by sending a message etc.)

2.13. Does the tool support sharing of project documents?

2.14. Does the system provide the authentication mechanism? (i.e. be able to use the system by login into the system)

2.15. Does the system provide the authorization mechanism? (i.e. giving the access rights to the different roles/people to reach the different parts of the tool)

2.16. Does the tool support usability in the distributed environment?

2.17. Does the tool have the ability to export/import data to/from external programs?

Table 1 gives the assessment results of project management tools. “N/A” fields in the table means “Not Applicable”.

Table 1 Project Management Tools Comparison

No	eProject	WebProject	SourceForge	MSProject	DSPMtool
1.1	**Yes**	**Yes**	**Yes**	**Yes**	**N/A** *(No information. Only the paper has been examined)*
1.2	**N/A** *(No information. Demo has been examined)*	**Largely** *(No information. Demo has been examined. But the use of Java significantly reduces the installation effort)*	**N/A** *(It requires Linux OS to run)*	**Yes**	**N/A** *(No information. Only the paper has been examined)*
1.3	**Yes**	**Yes**	**Yes**	**Yes**	**N/A** *(No information. Only the paper has been examined)*
1.4	**Yes**	**Yes**	**Yes**	**Yes**	**N/A** *(No information. Only the paper has been examined)*

Table 1 Project Management Tools Comparison (continued)

No	eProject	WebProject	SourceForge	MSProject	DSPMtool
1.5	Server (Windows) Clients (Platform independent)	Platform independent	Server (Linux) Clients (Platform independent)	Server (Windows) Clients (Platform independent)	Server (Windows) Clients (Platform independent)
1.6	**Yes**	**Yes**	**Yes**	**Yes**	**N/A** *(No information. Only the paper has been examined)*

Table 1 Project Management Tools Comparison (continued)

No	eProject	WebProject	SourceForge	MSProject	DSPMtool
2.1	**Partially** *(There is no detailed mechanism. Project planner creates the tasks, makes the assignments and sends them to the assignees. But any team member can create Tasks. The task creators are maintained as 'owners' and other Person/Users assigned or given access are 'Assignees')*	**Partially** *(WebProject offers 2 options: "Project Pinboards and Task Discussions", and "Chatter" (i.e virtual meeting room) to collaborate from the Project Planning task perspectives)*	**Partially** *(If the team member is assigned as Tool Administrator authorization, they can change the Topic/Title, assignments, dependencies, due date, work completed percent, and hours)*	**Partially** *(Workgroup members can create new tasks and send them to the project manager but they are not able to incorporate them into the project plan. Team leaders can make assignments)*	**No** *(Only project leader prepares the project schedule)*

Table 1 Project Management Tools Comparison (continued)

No	eProject	WebProject	SourceForge	MSProject	DSPMtool
2.2	**No** *(There is no a team preparing the project schedule but email notification is sent as determined by the Task Creator at task creation time, when assignment is completed)*	**Largely** *(by "Project Pinboards and Task Discussions", and "Chatter" (i.e virtual meeting room))*	**No**	**Partially** *(By email notification. But issues to be done for preparation of project plan is not informed to the project members. Workgroup members can not incorpaorat e their tasks into the project plan)*	**No**

Table 1 Project Management Tools Comparison (continued)

No	eProject	WebProject	SourceForge	MSProject	DSPMtool
2.3	**No** *(There are predefined roles: Team member, project manager. Personnels are created as either Project Manager or Team Member roles. This assignment is made within the User Management subsystem. But this 'role' assignment is of limited implications)*	***Partially*** *(There are predefined roles: Admin, Leader, Team member and User. There are various configuration codes that are used to define organizational elements into the system; the three primary codes are Group Codes, Resource Breakdown Structure (RBS) codes and Organizational Types. Also Project tasks can be saved to a Project template which allows re-use of the project structure)*	**No** *(Outside of the pre-defined roles, the functionality of organizational roles are not supported. And there is no organization of resources)*	**No** *(Outside of the pre-defined roles, there is no organizational level information definition)*	***Partially*** *(Only project members must first be defined in the organization)*
2.4	**No**	**Yes**	**No**	**Yes**	**No**

Table 1 Project Management Tools Comparison (continued)

No	eProject	WebProject	SourceForge	MSProject	DSPMtool
2.5	**No**	**Partially** *(There are various configuration codes that are used to define organizational elements into the system, the three primary codes are Group Codes, Resource Breakdown Structure (RBS) codes and Organizational Types. Also Project tasks can be saved to a Project template which allows re-use of the project structure)*	**No** *(Outside of the pre-defined roles, the functionality of organizational roles are not supported. And there is no organization of resources.)*	**No**	***Partially*** *(Only project members may be selected from the organization repository)*
2.6	**No**	**No**	**No**	**Partially** *(There exist teams and team leader, but not in a hierarchical structure)*	**Yes**

Table 1 Project Management Tools Comparison (continued)

No	eProject	WebProject	SourceForge	MSProject	DSPMtool
2.7	**No** *(Tasks may be assigned to ONE Person/User)*	**Partially** *(Tasks may be assigned to more than one person not to a team)*	**No** (*Tasks may be assigned to ONE person)*	**Yes**	**Yes**
2.8	**Partially** *(Teams are not assignable to tasks. But more than one task can be assigned to one person)*	**Partially** *(Teams are not assignable to tasks. But more than one task can be assigned to one person)*	**Partially** *(Teams are not assignable to tasks. But more than one task can be assigned to one person)*	**Yes**	**Yes**
2.9	**No** *(There is no mechanism for commitment, but Once a task is created, it is live in the system and the Task Creator and the Task Assignee can write 'progress notes' against it)*	**Largely** *(Team member can manipulate tasks that it is assigned, participate in discussions and chat, as well as configure it's own calendar)*	**No** *(Only If the team member is assigned as Tool Administrator authorization, they can change the Topic/Title, assignments, dependencies, due date, work completed percent, and hours)*	**No** *(There is no mechanism for commitment)*	**No**

Table 1 Project Management Tools Comparison (continued)

No	eProject	WebProject	SourceForge	MSProject	DSPMtool
2.10	**No** *(Project is defined only a Task List. There is no dependency between tasks)*	**Yes**	**Yes**	**Yes**	**No**
2.11	**No**	**No**	**No**	**Partially** *(There is no approval mechanism, but the master project plan Is under the project manager control. The others can only send their tasks to the project manager to incorporate them)*	**No**
2.12	**Yes** *(by means of email notification system)*	**Largely** *(by means of Project Message Board and Chatter)*	**No**	**Yes** *(by means of email notification system)*	**No** *(All tasks are made publicly viewable)*

Table 1 Project Management Tools Comparison (continued)

No	eProject	WebProject	SourceForge	MSProject	DSPMtool
2.13	**Yes** *(Document file sharing system provides privacy permission setting, and helpful file descriptors and folder storage)*	**No**	**No**	**No**	**Largely** *(Project documents are managed through the lifetime of the project)*
2.14	**Partially** *(In reporting terms, the user views are restricted by user log-on)*	**Yes** *(All users are required to login to the system for access)*	**Yes** *(All SourceForge users must log in to use the system)*	**Yes** *(Project Central supports both Windows NT Authentication and its own authentication scheme, which uses username/password pairs maintained by Project Central)*	**N/A** *(No information)*

Table 1 Project Management Tools Comparison (continued)

No	eProject	WebProject	SourceForge	MSProject	DSPMtool
2.15	**Partially** *(The Documents storage system supports access permissions by Person/User)*	**Yes** *(The user profile accessibility controls are used to allow/deny access to various aspects of the project as defined by the Administrator)*	**Yes** *(The project administrator can limit the visibility of the tools to only those that are designated as team members and can limit the accessability of project objects on a per team member basis)*	**Yes** *(The project administrato r determines what data will be available to each users)*	**N/A** *(No information)*
2.16	**Yes** *(it is a web-based tool)*	**Yes** *(it is a web-based tool)*	**Yes** *(it is a web-based tool)*	**Yes** *(it is a web-based tool)*	**Yes** *(supports distributed environment)*
2.17	**Partially** *(The transfer is one-way; export from the eProject system to MS Outlook client)*	**No**	**No**	**Yes**	**No**

WebProject is implemented in Java environment, therefore it is platform independent. Only the SourceForge's server is available in Linux operating system. The others' servers except WebProject's are available in Windows operating system. Clients of all tools are platform independent.

None of the tools covers all functional attributes. All tools except DSPMtool partially supports preparation of project plan collaboratively but there is no well defined mechanism for colloboration. For example MS Project with Project Central enables the workgroup members to create new tasks but they can not incorporate them into the project plan. They should send them to the project manager for incorporation.

None of the tools, except DSPMtool and WebProject, enable definition of organizational level information: personnel, role, activity. Therefore, they don't offer the use of the organizational level information while preparaing the project plan. But DSPMtool supports only personnel definition at the organization level and usage of them in the project plan preparation. WebProject enables to save project tasks to a Project template which allows re-use of the project structure.

Only MS Project with Project Central and DSPMtool supports team definition. But only the DSPMtool supports definition of teams in a hierarthical structure. Therefore only MS Project and DSPMtool enable assignment of tasks to teams.

Only WebProject supports partially commitment of task assignments by means of discussion and chat mechanism. Three of the tools, WebProject, SourceForge and MS Project enables the activity dependency definition. Only MS Project with Project Central supports partial approval of project plan. In a way that the master project plan is under the control of project manager, the other members should send their tasks to incorporation into the project plan. Three of the tools, eProject, WebProject and MS Project provide some kinds of to do list by email notification or message board mechanism. DSPMtool makes all tasks publicly viewable.

Only eProject and DSPMtool offers document sharing mechanism. Almost all tools, except DSMtool (there is no information for DSPMtool), supports authentication and authorization mechanism. Only MS Project has the ability to

export and import data to/from external programs. And eProject supports exporting data to MS Outlook client.

2.5 Summary

In this chapter, project management concept has been explained. Especially, the focus is given to project planning process. To understand the project planning, related knowledge areas defined in the PMBOK and the construction of WBS has been mentioned. Then the need of automation support to project planning has been revealed. Moreover, existing project management support tools have been introduced and a comparison of these tools has been performed according to the provided checklist. This section ends with the results of this comparison.

CHAPTER 3

DSPPTool ANALYSIS AND DESIGN

In the first section, the requirement analysis of the Distributed Software Project Planning Tool (DSPPTool) is presented. Use case method is used for elicitation of the requirements of the DSPPTool. Use cases are grouped under the related perspectives and briefly described.

In the second section, the design of DSPPTool based on the requirements is presented. Object-oriented methodology is used also in design of DSPPTool. Firstly, modules of the DSPPTool is given and explained in the Module Decomposition section. Also classes of modules that realize the use cases identified in the requirement analysis are explained. In the Data Decomposition section, the database tables used in each module is given and explained. There are two groups of interfaces. Graphical User Interfaces (GUI) and an interface with the MS Project 2000. In the Interface Description section, these interfaces are explained.

3.1 DSPPTool Analysis

3.1.1 Introduction

The purpose of this chapter is to present the software requirements analysis of the DSPPTool by using Unified Modeling Language (UML) with the support of graphical representations.

The DSPPTool shall meet both functional and operational requirements stated in the Section 2.4.

3.1.2 General Description

This section represents general factors that affect DSPPTool requirements.

3.1.2.1 Product Perspective

The DSPPTool shall be developed by using Java (SDK 1.3 distribution). As all know, java provides platform independency, this independency provides the widespread use of the tool. Moreover, to enable the tool run in a distributed environment, RMI technology shall be used.

The underlying database management system shall be Microsoft Access, which seems to cause difficulty in porting the DSPPTool to other operating systems. However, Java Database Connectivity (JDBC), which shall be utilized to establish a bridge to access database tables, will provide methods for changing the underlying database management system without too much effort. The layered architecture of the DSPPTool shall separate the database layer from bussiness and interface layers by using Database Access Objects (DAO) patterns, therefore it provides an effective way to port the system to other environments. The only part to be updated is only one DAO class.

3.1.2.2 Product Functions

The actors of the DSPPTool are given in the Figure 4.

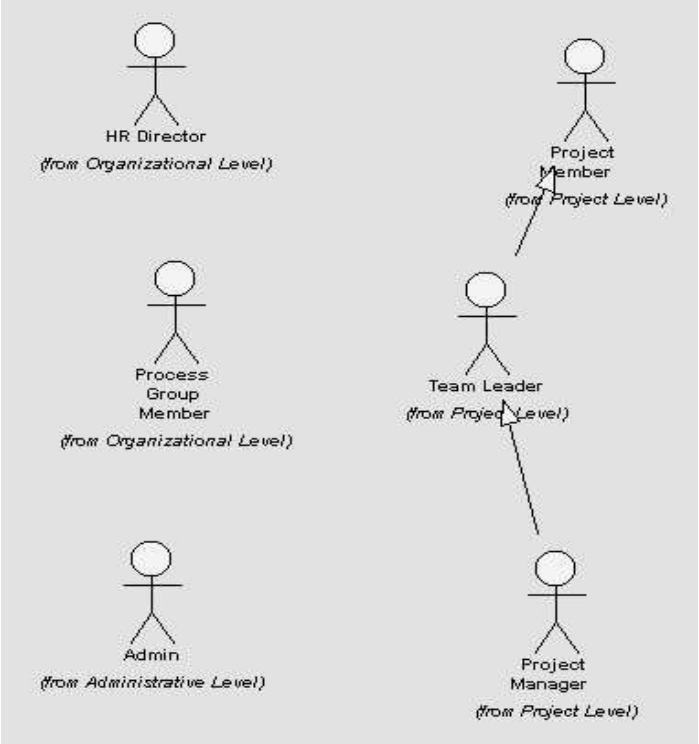

Figure 4 Actors of DSPPTool

- ***HR Director:*** He/she uses the system for the organizational level needs: to add/delete personnel and to update the personnel information.
- ***Process Group Member:*** He/she uses the system for the organizational level needs: to add/delete role or activity and to update the role or activity information.
- ***Project Member:*** Person who is the member of the project. He/she uses the system to make committed/uncommitted the activities assigned to him/her by the team leader/project manager. Also he/she accesses the To Do List including all activities assigned to him/her by using the system.
- ***Team Leader:*** Person who is the leader of one of the teams in the project. This actor is inherited from the Project Member actor. Therefore he/she may use the system for the same purpose of the Project Member actor. In addition, he/she uses the system to participate in preparation of project schedule.

- ***Project Manager:*** Person who is the manager of one project in the system. This actor is inherited from the Team Leader actor. Therefore he/she may use the system for the same purpose of the Team Leader actor. In addition, he/she uses the system to define a new project with the project members' and teams' definitions, delete a project from the system and update the project information with the all details and to participate in preparation of project schedule.
- ***Admin:*** Person who is the administrator of the system. He/she uses the system to define users of the system, delete the user from the system and update the user information. In addition, he/she may define the executable programs by using the system.

The functions of the DSPPTool can be grouped into three:

- ***Organizational Knowledge Board Operations:*** In this group, the functions related with organizational level information take part. These are *Personnel Definition, Role Definition* and *Activity Definition* -in which activities are defined in a hierarchical structure. The related functions can be seen in the Figure 5.

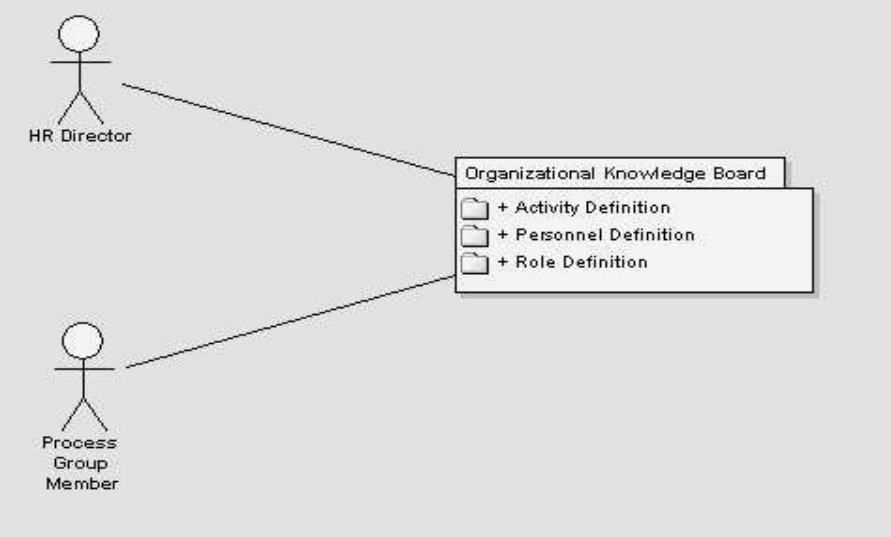

Figure 5 Organizational Knowledge Board Operations

- ***Project Knowledge Board Operations:*** Project level functions occupy in this group. The related functions can be seen in the Figure 6.These are:
    - *Project Definition:* in which project general information is defined, project members are identified with their roles by selecting from the organizational personnel and role pool, teams are constructed in a hierarchical structure by selecting the personnel–role mapping-
    - *Project Schedule Operations:* in which project schedule is prepared by utilizing the activities from the organizational activity pool and by assigning these activities to the project members or a project team. Also To Do List including the issues to be done while preparing the project schedule is presented to the project manager and the team leaders. In addition, To Do List including all activities assigned to a project member is presented to this project member when logging in to the system.
    - *Project File Operations:* in which project documents are defined by selecting the documents from the file system, access rights of the project members to the project documents are identified.

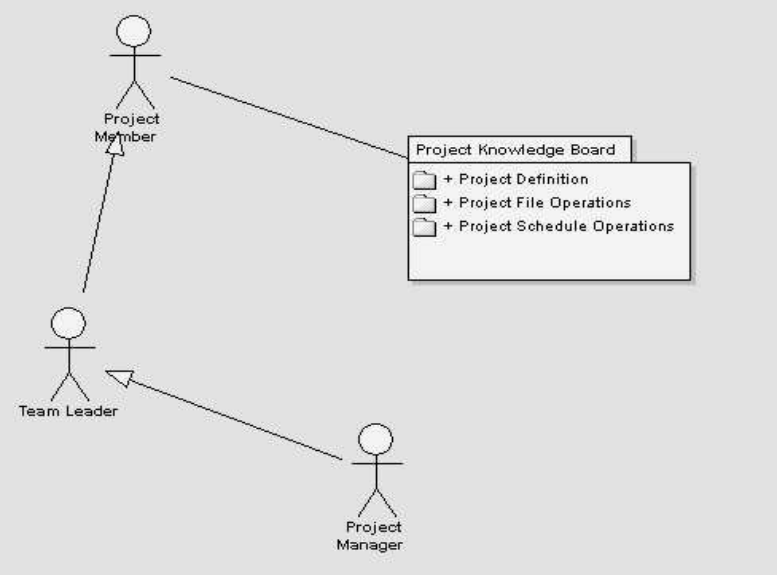

Figure 6 Project Knowledge Board Operations

- ***Administrative Operations:*** The following functions exist in this group: *User Definition* and *Executable Programs Definition* in which executable programs to be utilized to open the project documents are defined in the system by selecting the executable files from the file system. These functions can be seen in the Figure 7.

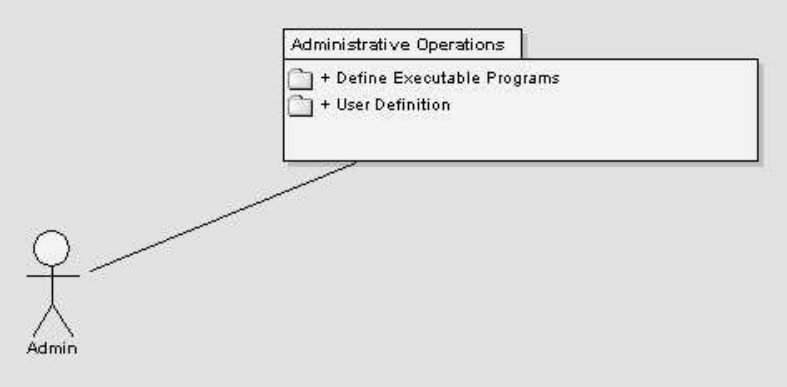

Figure 7 Admin Operations

3.1.2.3 Constraints, Assumptions and Dependencies

The following assumptions have been made for the requirement specification.

- It is assumed that each project team shall have one team leader.
- It is assumed that users of DSPPTool have knowledge of software project planning

3.1.3 Specific Requirements

This section presents all of the specific requirements of DSPPTool. Functional requirements are emphasized and explained as groups of related use cases. For the details, see [16].

3.2 DSPPTool Design

3.2.1 Decomposition Description

3.2.1.1 Module Decomposition

Identified main modules of DSPPTool are Organizational Knowledge Board, Project Knowledge Board, Administrative Operations and Utility modules. See the Figure 8 and [16] for the details.

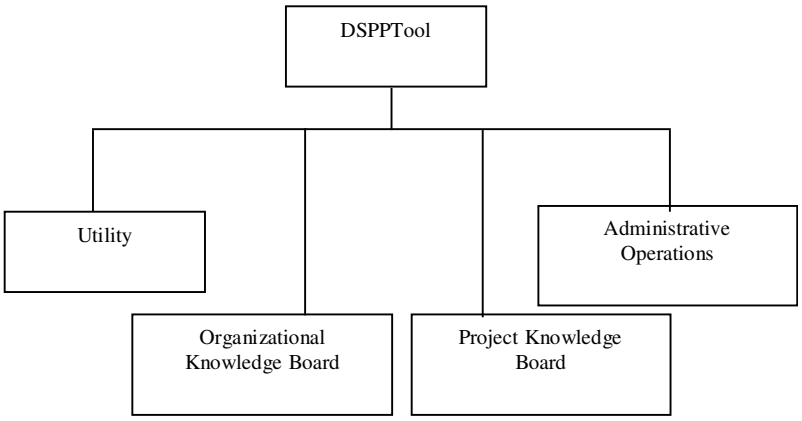

Figure 8 Modules of DSPPTool

3.2.1.2 Data Decomposition

The following database tables are designed: For the details, see [16].

- **ACTIVITIES:** This table keeps activity information.
- **ACTIVITYROLES:** This table keeps the activity role pair in which role is able to perform the activity.
- **EXECUTABLEPROGRAMS:** This table keeps the executable program information.
- **FILEACCESSRIGHTS:** This table keeps the access rights of the personnel to a project document.
- **FILES:** This table keeps the project document information.

- **PERSONNELROLES:** This table keeps the personnel role pair for a project.
- **PERSONNELS:** This table keeps the personnel information.
- **POSTWBSITEMS:** This table keeps the post WBS items.
- **PREWBSITEMS:** This table keeps the pre WBS items.
- **PROJECTS:** This table keeps the project information.
- **ROLES:** This table keeps the role information.
- **TEAMPERSONNELS:** This table keeps the personnel of a team.
- **TEAMS:** This table keeps the team information.
- **USERS:** This table keeps the user information.
- **WBSITEMASSIGNMENTS:** This table keeps the assignment information of a WBS item.
- **WBSITEMFILES:** This table keeps the documents of a WBS item.
- **WBSITEMS:** This table keeps the WBS item information.
- **WBSITEMTOBEPRODUCEDFILES:** This table keeps the to be produced documents of a WBS item.
- **WBSPARAMETERS:** This table keeps the WBS parameters of a project.
- **WBSVERSIONCOMMITMENTLIST:** This table keeps the commitment information of a person for a WBS version.
- **WBSVERSIONS:** This table keeps the WBS version information.

3.2.2 Interface Description

There are two groups of interfaces: Graphical User Interfaces (GUI) and an interface with the MS Project 2000.

3.2.2.1 Interface with the MS Project 2000

MS Project 2000 has a capability of importing a text file with the predefined format. DSPPTool is integrated with MS Project 2000 in a way that it has a capability of exporting the prepared project schedule as a text file with the predefined format specified by MS Project 2000. This text file can be imported into MS Project 2000. By means of this integration, project schedule data (WBS item name, start and finish dates, WBS item dependencies, assigned people name,

surname and their percentage, assigned team name) created in the DSPPTool is transfered to the MS Project 2000 and then the Gannt chart view of the project schedule may be seen.

Since the predefined format of the MS Project does not support to display the WBS Items in a tree structure, the tree structure of the WBS items in the exported project schedule from DSPPTool can not be preserved. But since the order of WBS items are preserved, by giving the name showing the hierachy to the WBS items may provide the tree structure in a visual way.

3.2.2.2 Graphical User Interfaces

Graphical user interfaces establish a communication with the users. For the details, see [16].

CHAPTER 4

EVALUATION OF THE TOOL

The tool evaluation is based on two criteria, First one is the tool's adequacy to meet the functional attributes introduced in Chapter 2. The second one is the validation of the tool by utilizing the data of the project schedule of a real project provided by MilSOFT A.Ş. The compliance to the first criteria will be discussed in the following section. The second section presents validation results. In the third section, how the tool supports the knowledge areas of PMBOK described in the Chapter 2 is explained.

4.1 Meeting the Operational and Functional Attributes

Table 2 presents the comparison of the system's functionality with respect to the attributes defined in Chapter 2. According to this table, the DSPPTool meets the 90% of the listed attributes. Detailed explanations are given in the Comments column of the table. Table 3 presents the summary comparison results of the tools presented in the section 2 and our tool, together.

Table 2 Meeting the Operational and Functional Attributes

No	DSPPTool	Comments
1.1	**No**	It is not in the scope of the thesis
1.2	**Largely**	Only server installation required. Clients do not need any installation. They can use the application from their browsers which support java applet. On the server, JRE 1.3 should be installed and RMI should be run. Web server configuration for entrance html file which contains applet should be performed.
1.3	**No**	It is not in the scope of the thesis
1.4	**No**	Since there is no user manual.
1.5	**Platform Independent**	The system developed in Java environment therefore it is platform independent
1.6	**Yes**	
2.1	**Yes**	The DSPPTool is developed with the multi-user support
2.2	**Yes**	Team leaders are informed about issues to be done to prepare the project schedule by means of Project Schedule To Do Lists
2.3	**Largely**	In the organizational level, personnel, roles and activities are defined.
2.4	**Yes**	Activities are defined in a tree structure.
2.5	**Largely**	In the definition of project, project members are selected from the organizational personnel pool and their roles are defined by selecting from the organizational role pool. In the preparation of the project schedule, activities are selected from the organizational level activity pool. Selected parent activity is transferred with its children to the project schedule.
2.6	**Yes**	Project Teams are defined in a tree structure.
2.7	**Yes**	One activity can be assigned to one or more project members or a project team.
2.8	**Yes**	More than one activity can be assigned to a project member or to a project team.
2.9	**Yes**	All project members can specify their commitment information by marking the project schedule as committed or uncommitted
2.10	**Yes**	Both pre and post activities of an activity can be defined
2.11	**Yes**	Project Manager approves the project schedule with SUBMITTED status. After approval, the status of the project schedule becomes APPROVED.
2.12	**Yes**	Project members are informed about their assigned tasks by means of Project Assignment To Do Lists
2.13	**Partially**	The DSPPTool gives the required information, such as the file path of documents to be used and document names of documents to be produced, to access these documents. Creation and opening of the documents are left to the executed programs and the user responsibility.
2.14	**Yes**	Each user shall be log into the system to use the DSPPTool
2.15	**Yes**	The DSPPTool provides the access rights to the different roles to reach the different parts of the tool. For example, only the project managers are able to approve the project schedule, only the team leader can specify a WBS item which is assigned to the team leader's team, etc.

Table 2 Meeting the Operational and Functional Attributes (continued)

No	DSPPTool	Comments
2.16	**Yes**	The DSPPTool is developed with the multi-user support
2.17	**Partially**	The DSPPTool has a capability of exporting the prepared project schedule as a text file. This text file can be imported into MS Project 2000. By means of this integration, project schedule data (activities, start and finish dates, activitiy dependencies, assigned people and their percentage) created in our tool is transfered to the MS Project and then the Gannt chart view of the project schedule may be seen. Since the predefined format of the MS Project does not support to display the WBS Items in a tree structure, the tree structure of the WBS items in the exported project schedule from DSPPTool can not be preserved.

Table 3 Project Management Tools Comparison including DSPPTool

No	DSPPTool	eProject	WebProject	SourceForge	MSProject	DSPMtool
1.1	No	Yes	Yes	Yes	Yes	N/A
1.2	Yes	N/A	Largely	N/A)	Yes	N/A
1.3	No	Yes	Yes	Yes	Yes	N/A
1.4	No	Yes	Yes	Yes	Yes	N/A
1.5	Platform Independent	Server (Windows) Clients (Platform independent)	Platform independent	Server (Linux) Clients (Platform independent)	Server (Windows) Clients (Platform independent)	Server (Windows) Clients (Platform independent)
1.6	Yes	Yes	Yes	Yes	Yes	N/A
2.1	Yes	Partially	Partially	Partially	Partially	No)
2.2	Yes	No	Largely	No	Partially	No
2.3	Yes	No	*Partially*	No	No	*Partially*
2.4	Yes	No	Yes	No	Yes	No
2.5	Yes	No	Partially	No	No	*Partially*
2.6	Yes	No	No	No	Partially	Yes
2.7	Yes	No	Partially	No	Yes	Yes
2.8	Yes	Partially	Partially	Partially	Yes	Yes
2.9	Yes	No	Largely)	No	No	No
2.10	Yes	No	Yes	Yes	Yes	No
2.11	Yes	No	No	No	Partially	No
2.12	Yes	Yes	Largely	No	Yes	No
2.13	Partially	Yes	No	No	No	Largely
2.14	Yes	Partially	Yes	Yes	Yes	N/A
2.15	Yes	Partially	Yes	Yes	Yes	N/A
2.16	Yes	Yes	Yes	Yes	Yes	Yes
2.17	Yes	Partially	No	No	Yes	No

4.2 Validation of the Tool by Using Project Schedule Data

4.2.1 MilProject Characteristics

General characteristics of MilProject can be summarized as follows:

- Development life cycle is Waterfall.
- Number of project members is 10.
- Number of project teams is 5. These are:
    - *Requirement Team:* This team is responsible for requirement specification activities. It contains 4 members with the following roles: analyst, project manager, software engineer and quality assurance specialist.
    - *Design Team:* This team is responsible for design activities. It contains 3 members with the following roles: architect, designer and coder.
    - *Implementation Team:* This team is responsible for implementation and unit test activities. It contains 3 members with the following roles: coder and tester.
    - *Test Team:* This team is responsible for unit integration and testing and software acceptance test activities. It contains 4 members with the following roles: project manager, tester, software engineer and quality assurance specialist.
    - *Training Team:* This team is responsible for training activities. It contains 3 members with the following roles: analyst, software engineer and coder.
- Project start and finish dates are 03.01.2000 and 09.06.2000 respectively.
- Total effort spent is approximately 50-person month.

4.2.2 Project Plan Preparation for MilProject using DSPPTool

Project plan by using DSPPTool was prepared by following the steps given below:

1. Firstly by using Organizational Knowledge Board capabilities;
    1.1. Personnel were identified by means of Personnel List GUI.

1.2. Roles were defined at the organizational level by means of Role List GUI. The roles can be seen in the Figure 9.

1.3. The activities in the project schedule of MilProject were defined as the organizational level activities. The activities can be seen in the Figure 10.

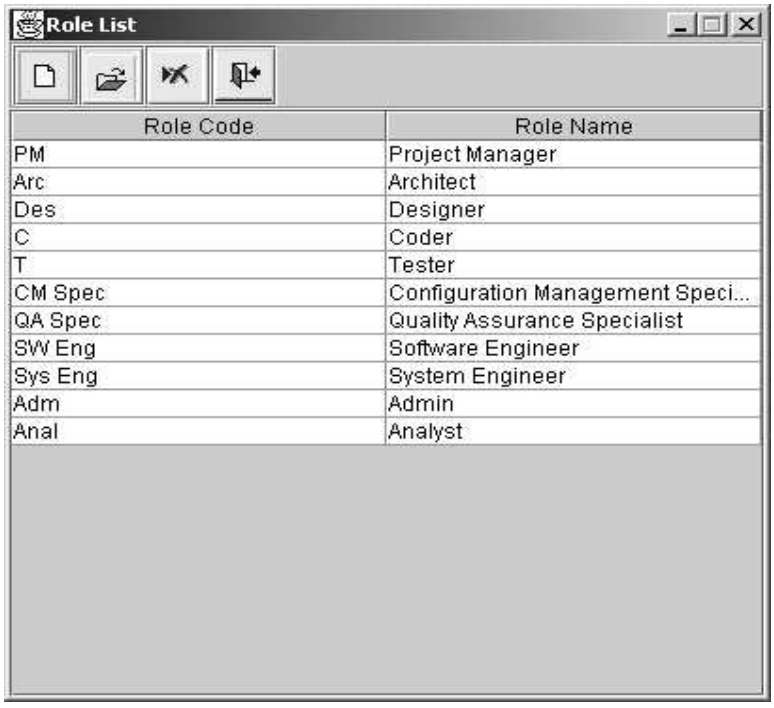

Figure 9 Organizational Role List

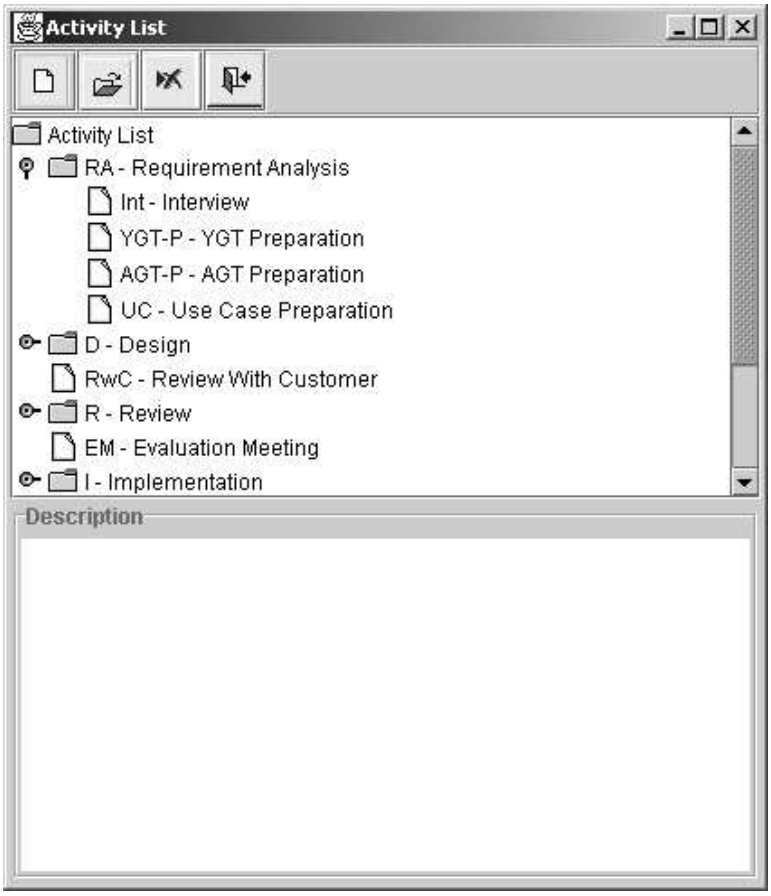

Figure 10 Organizational Activity List

2. By using Project Knowledge Board capabilities;
   2.1. Project was defined with its project members and their roles by utilizing the organizational level information. This can be seen in the Figure 11 and Figure 12.

Figure 11 Project Member-Role Selection (by using Organizational Information)

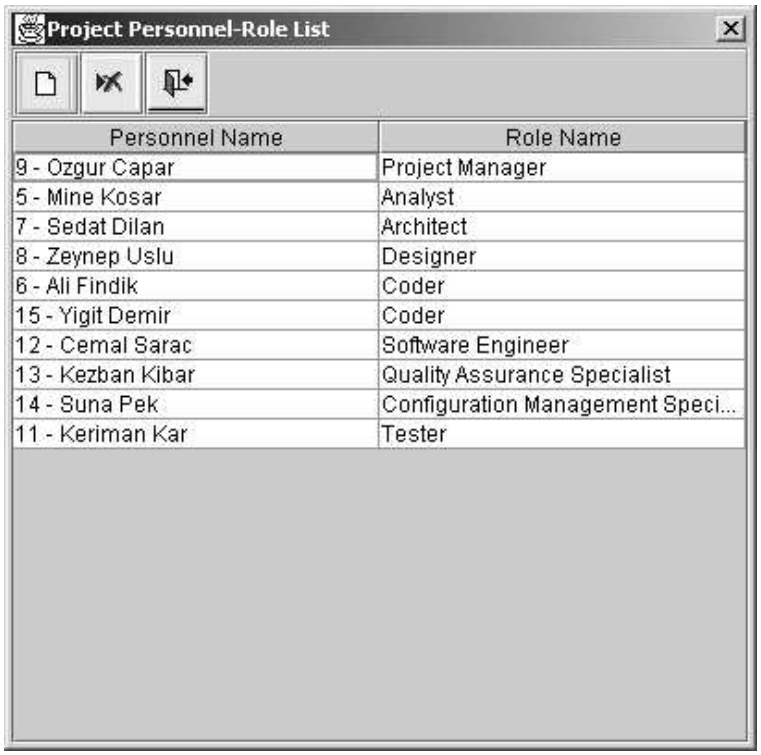

Figure 12 Project Personnel-Role List

2.2. Project teams were constructed. The entered teams can be seen in the Figure 13 and Figure 14.

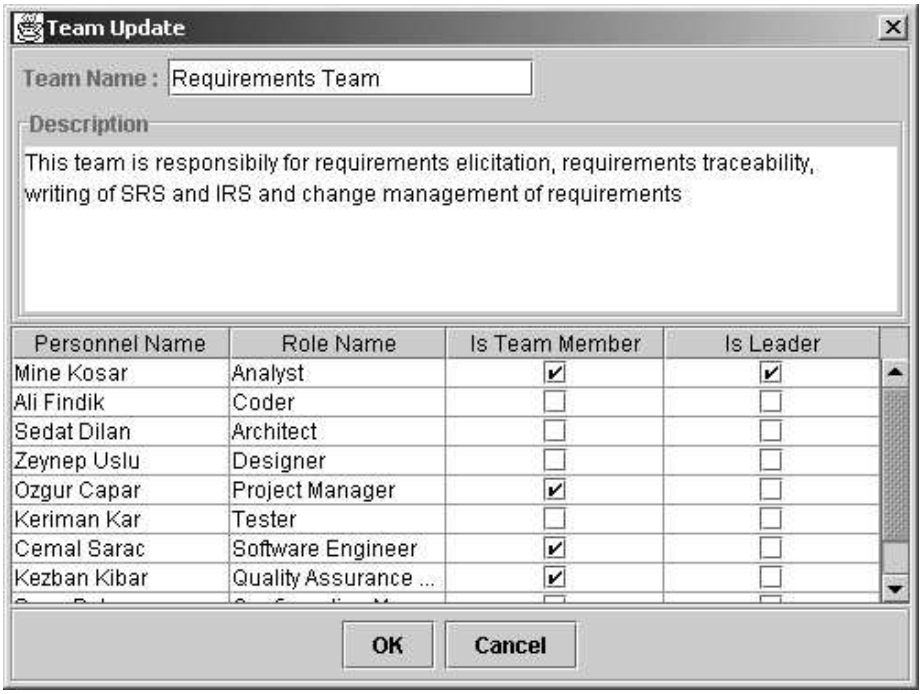

Figure 13 Team Definition

Figure 14 Team List of MilProject

2.3. To prepare project scope, logged into the system as project manager.

2.4. High-level WBS items were defined by means of WBS item GUI.

2.5. While preparing the MilProject scope and schedule, activities were selected from the organizational pool. This can be seen in the Figure 15.

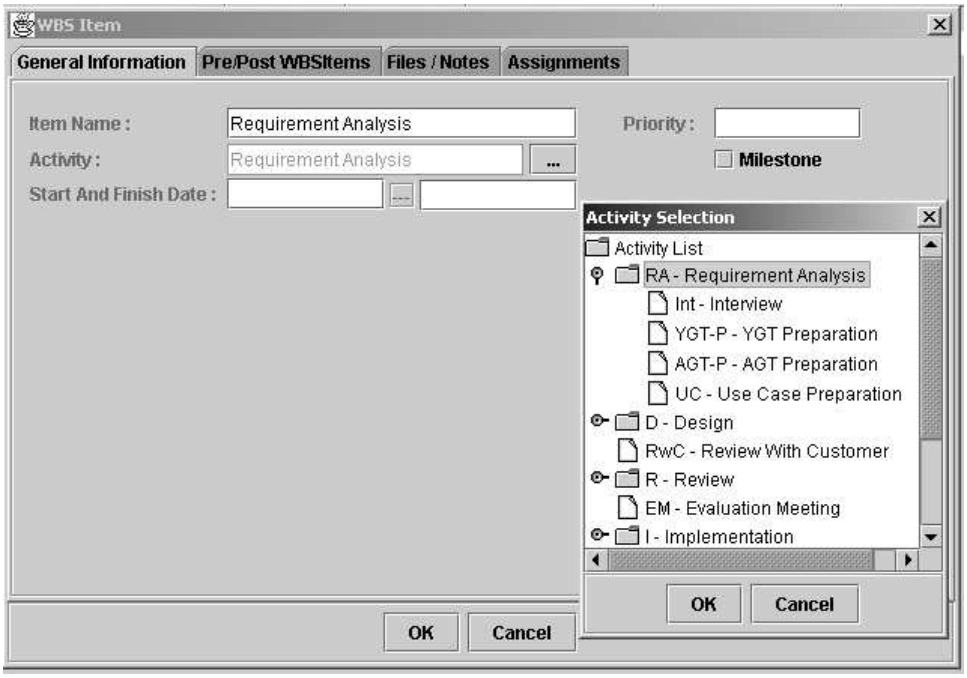

Figure 15 Selecting Organizational Level Activities

2.6. Selected organizational level activity is transferred to the project plan with its children (See Figure 16).

2.7. Each high level WBS item was assigned to a project team (Figure 17).

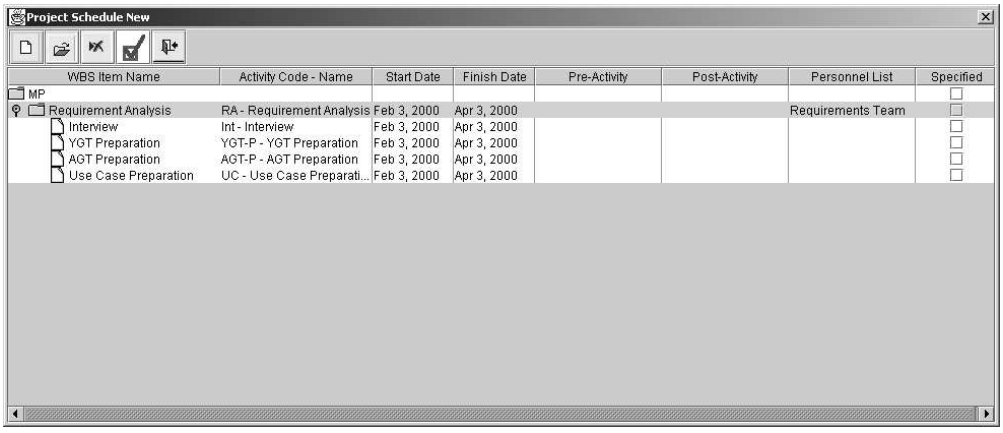

Figure 16 Requirements Analysis Activity with its Children

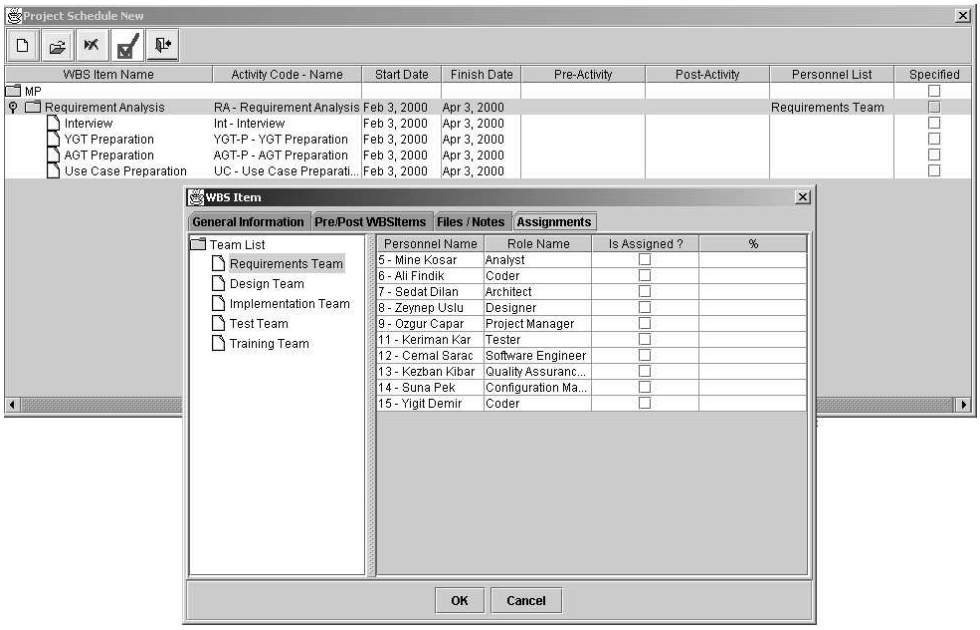

Figure 17 Requirement Analysis Activity Assignment to the Requirements Team

2.8. After the definition of the high level WBS items. Each team leader should detail the WBS item assigned to their team. At that point, each team leader was informed about this issue by means of Project Schedule To Do List (Figure 18).

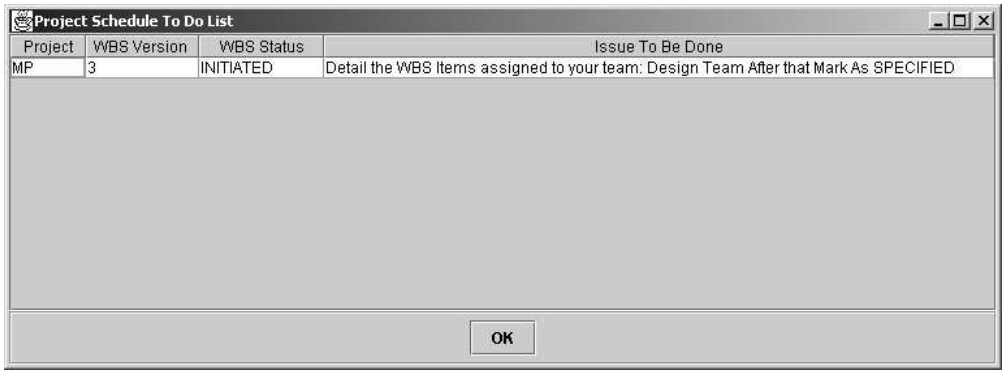

Figure 18 Project Schedule To Do List

2.9. To detail project scope and schedule, the user logged into the system as team leader.

2.10. To specify the WBS item, the team leader created the detailed level WBS items if required

2.11. The team leader assigned the WBS items to the team members.

2.12. He/she marked the WBS items as specified.

2.13. Each team leader performed the steps from 2.8 to 2.11

2.14. Project manager made the same thing with the step 2.11 for the project, the project schedule status became SUBMITTED.

2.15. At that point, each project member was informed about issue to be done for entering his or her commitment by means of Project Schedule To Do List. Each project member may have identified his or her commitment as “committed” or “uncommitted”.

2.16. When the project schedule is in SUBMITTED status, the Project Manager should approve the project to complete the project schedule preparation.

2.17. After approval of the schedule, its status became APPROVED. The preparation of project scope and schedule was completed. The project schedule can be seen in the Figure 67. Note that the completed items can be distinguishable with the different color.

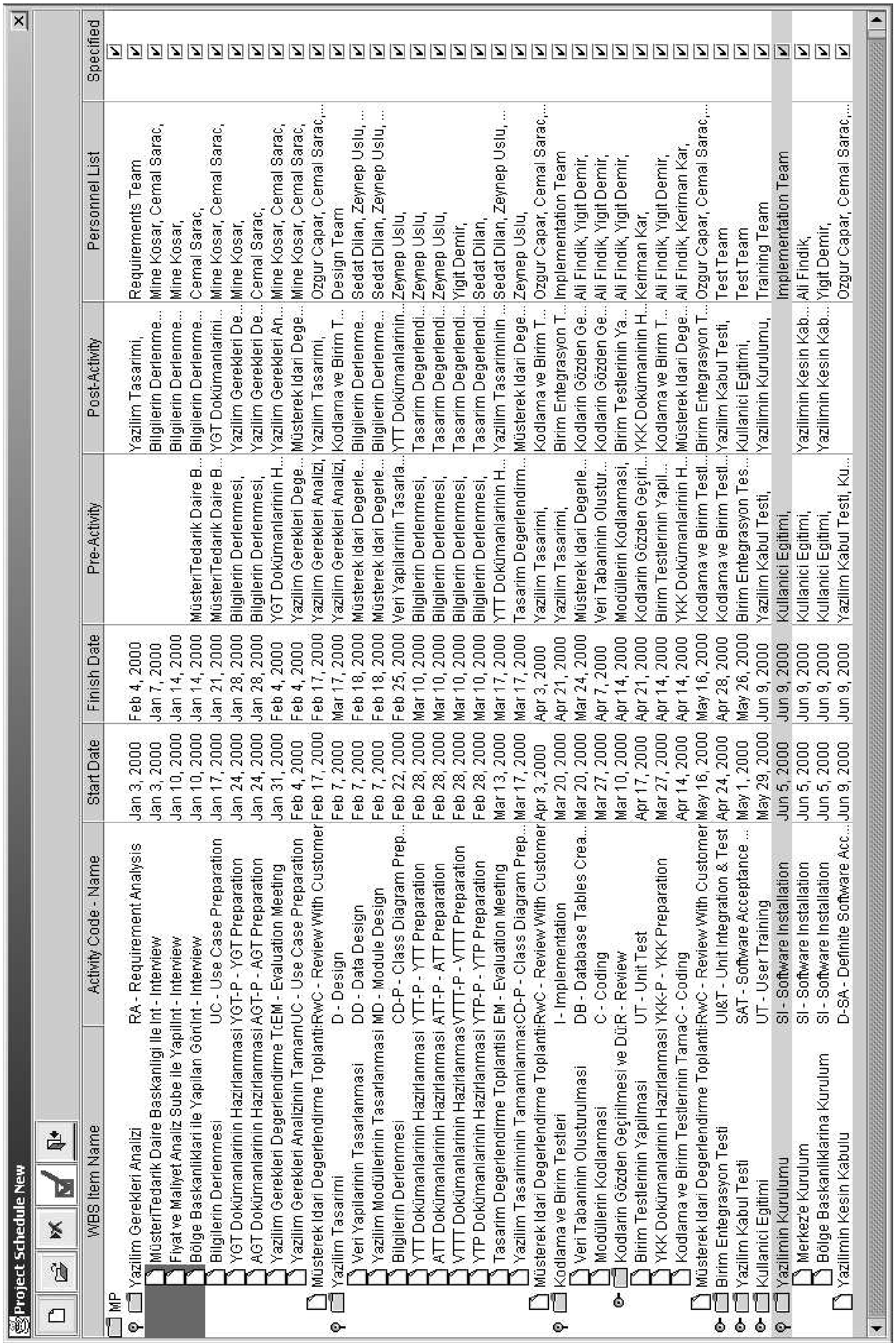

Figure 19 Project Schedule of MilProject

With the DSPPTool, described personnel, roles and activities can be utilized by the other projects other than MilProject. This provides the reusability of defined information, consistency of information throughout the organization and prevents the misunderstanding caused by recreation of information in different projects.

While defining the high level WBS item, activity selected from the organizational pool may be a parent activity. In that case, parent activity is transferred to the project schedule with its children. This may be advantage because you don't have to recreate the activities. But it may be a disadvantage if you want to create only the parent activity. In that case, team leader should delete the children of the parent activity. Because only the team leaders may change the details of the WBS items assigned to their team. But this case can be removed in a way that before making the assignment the children of the parent activity can be deleted by the project manager.

As seen in the steps given above, after the definition of the high level WBS items. Each team leader should detail the WBS item assigned to their team. The team leaders learn their issues to be done for the specification of the WBS items by means of their Project Schedule To Do List. When the user logs into the system, the Project Schedule To Do List and Project Assignment To Do List are displayed. In this way the team leader knows that he/she should specify the WBS items assigned to his/her team. The mechanism makes the project members participate into project schedule preparation.

After approval of the project scope and schedule, each project member is informed about their assignments by means of Project Assignment To Do List. During the development of the project, the project members are able to use their Project Assignment To Do List to remember their assignments. As the project members complete their assignments, they should enter the actual start and finish date of their assignments and marks them as COMPLETED in the Project Assignment To Do List. Project manager and team leaders can be aware of the completed activities in the project schedule view by means of the different coloring of the completed activities. To change the color of a WBS item for showing the completion, all project members who are assigned to this WBS item

should mark this WBS item as COMPLETED. This capability contributes to the project control.

Since the software is intangible and there may exist many uncertainties in the software requirements, the whole project schedules may not be prepared in detail at the start of the project. In that case, the some parts of project schedule for the closer days are prepared more detailed. And as time goes, the project schedule is updated to make it more detailed. For such a case, the DSPPTool provides the capability of "Save As New Version". By creating the new version of approved project schedule in that way, a new version of the schedule contains the same items with the INITIATED status.

According to the explained observations gained by means of utilizing a real project schedule data, the tool can be identified as a successful product. However, these cannot be considered as statistically significant results. The tool should be used for more real projects even in different organizations. Within our limitations, the tool can be classified as a successful tool to support the software project planning process.

4.3 Supporting of Knowledge Areas of PMBOK

Table 4 presents how the tool supports the knowledge areas of PMBOK described in the Chapter 2. Detailed explanations are given in the "How DSPPTool Supports" column of the table.

Table 4 Supporting of Knowledge Areas of PMBOK

Knowledge Area	Process	How DSPPTool Supports
Project Scope Management	Initiation	DSPPTool offers a simple mechanism to initiate a project. Defined projects in the tool are assumed that they are initiated ones.
	Scope Planning	DSPPTool does not support scope-planning process.
	Scope Definition	DSPPTool supports scope definition process in a way that high level WBS of a project can be defined.
	Scope Verification	DSPPTool does not support scope verification process.
	Scope Change Control	DSPPTool supports scope change control process in a way that new versions of project WBS can be created and the previous versions are kept in the tool.
Project Time Management	Activity Definition	DSPPTool fully supports activity definition process in a way that high level WBS items are decomposed into smaller, more manageable components. These components are activities to be performed to complete the project, which are defined at the organizational level.
	Activity Sequencing	DSPPTool supports activity sequencing process in a way that pre and post activities of an activity can be defined.
	Activity Duration Estimating	DSPPTool does not support activity duration estimating process.
	Schedule Development	DSPPTool offers ability of entering and keeping start and finish dates of activities. Also it supports assignment of activities to the project members/team. Thus project schedule becomes prepared.

Table 4 Supporting of Knowledge Areas of PMBOK (continued)

Knowledge Area	Process	How DSPPTool Supports
	Schedule Control	DSPPTool supports schedule change control process in a way that new versions of project schedule can be created and the previous versions are kept in the tool.
Project Human Resource Management	Organizational Planning	DSPPTool supports organizational planning process in a way that roles and personnel are defined at the organizational level., project members are identified with their roles by selecting personnel and his/her role from the organizational pool. But it does not support staff management planning, which describes when and how human resources will be brought onto and taken off of the project team.
	Staff Acquisition	DSPPTool does not support staff acquisition process.
	Team Development	DSPPTool only offers ability of constructing project teams and making assignments to those project teams. But it does not support other aspects related with psychological and managerial issues of team development.

CHAPTER 5

CONCLUSION AND FUTURE DIRECTIONS

This chapter concludes the thesis by determining how the developed tool fulfilled the objectives and the aim identified in Chapter 1. The results of the tool evaluation (explained in the Chapter 5) made by utilizing the data of MilProject enable to form a more realistic basis for these achievements. After the fullfilment of objectives and aim section, the conclusion section presents the significance and contribution of this thesis. And this chapter will end pointing out the future directions of work related to the studies on the software project planning automation support.

5.1 Fulfillment of Objectives and Aim

In the Chapter 1 three objectives are identified for DSPPTool:

- **Objective 1:** To provide a tool to support distributed software project planning
- **Objective 2:** To enable the use of the predefined organizational level information while preparing the project schedule
- **Objective 3:** To enable the effective use of the project documents through the project life cycle

With the DSPPTool, project schedule is prepared by cooperation of the project manager and the team leaders. Firstly project should be defined in the system by the project manager. In the project definition process, project general information is defined, project members are identified and project teams are constructed. After the project definition, Project Manager creates a project schedule with

INITIATED status. In the project schedule, he/she creates high level WBS items and assignes them to a team or one or more project member(s). WBS Items assigned to a team are detailed by the team leaders in a way that the detailed WBS items are created and assigned to the team members. After detailing WBS items, each team leader marks the WBS items assigned to him/her as SPECIFIED. Also Project Manager marks the project as SPECIFIED. At that point, the project schedule becomes SUBMITTED status and waits for the approval of the Project Manager. Before the approval, all project individuals may commits or uncommits the current version of project schedule. After the approval, the project schedule becomes APPROVED status. From that point, project schedule can not be changed. And all project members are able to see their assignments by means of their project assignment to do lists.

As it is explained in the previous paragraph, high level project scope, schedule and task assignments are prepared by the Project Manager, then high level items are detailed by assigned team leaders and finally all project members commit their assignments. Therefore all project individuals are made to participate in project planning process and objective 1 was fulfilled.

In the organizational level, personnel and roles are defined to be used while defining the project. Moreover activities are defined in a tree structure to be used while preparing the project schedule. While defining projects, project members are selected from the organizational level personnel pool and their roles are identified by selecting roles from the organizational level role pool. While preparing the project schedule, activities to be performed are selected from the organizational level activity pool. A parent activity is transfered with its children to the project schedule. This capability prevents re-considering and re-writing of the acivities in preparation of each proejct schedule. Also it gets rid of inconsistent definitions of activities. Thus organizational level information (personnel, role and activity information) is used while preparing the project schedule, in this way objective 2 was fullfilled.

Objective 3 was covered by the following capabilities with some limitations explained in here. While preparing the project schedule, documents to be used and

documents to be produced are identified. And the project individuals can see these documents by means of their project assignment to do lists. To enable individuals to access the project documents, DSPPTool provides required information, such as file path and document name. Also DSPPTool provides execution of required program to open documents. Creation and opening of these documents are left to the executed programs and user.

The aim identified in the Chapter 1 is as follows:

- **Aim:** To develop a distributed project planning tool that enables more than one people prepare the different parts of the project scope, schedule and task assignment by allowing the use of the predefined organizational level activities and enabling the effective use of the project information through the project life cycle.

It can be declared that this aim was successfully fulfilled through observation of conformance to the objectives.

5.2 Conclusion

This thesis focused on the development of a tool to support distributed software project planning process. DSPPTool provides the following capabilities:

- Definition of personnel, roles and activities at the organizational level
- Construction of project teams in a hierarchical structure
- Cooperation of project members to prepare the project scope, schedule and task assignments by using organizational level information
- Providing Project Schedule To Do List for notifying duties about project plan preparation
- Ability to assign tasks to a team or one or more project members
- Commitments of assignment by the project members
- Sharing of project documents throughout project life cycle
- Providing Project Assignment To Do List for notifying assigned tasks

By utilizing these capabilities, as given in Table 2 and Table 3, project planning problems given in detal in the Chapter 1 are overcomed. Therefore, DSPPTool makes a valuable contribution to project planning process.

As presented in the Chapter 5, in aspects of definition of organizational level information, preparation of project scope and schedule colloboratively by utilizing the organizational information and commitment of task assignments, the DSPPTool comprises more extensive capabilities than the tools presented in the Chapter 2. Therefore, this thesis is a significant study because of its contribution to reveal the deficiencies of existing project planning tools in aspect of colloborative preparation of project plan and utilizing the organizational information. As a conclusion, this thesis may be a reference to improve the capabilities of project planning tools.

5.3 Future Directions of Work

This study results in a tool that satisfies the basic requirements of automated support for the software project planning process and provides some facilities for practical use of organizational level information and for effective sharing of project documents.

The tool may be improved in aspect of sharing project documents by integrating with the commonly used Configuration Management tools.

Moreover in the organizational level, improvement to provide more detailed activity structure for the purpose of utilizing in the project schedule preparation may be performed.

Implementation of policy parameters of the projects defined in [16] may increase the usability of the tool. According to the policy parameters the mechansim of project schedule preparation may be done flexible:

- If the “Project Schedule may be approved only after all commitments have been completed” option is selected, approval of the project schedule is bound to that condition. Otherwise, project schedule can be approved without considering that condition.
- If the “Notify all members if a new version of the approved project schedule is created” option is selected, as said when the approved project schedule is saved as a new version, all project members should be notified by means of Project Schedule To Do List.

- If the “Allow Project Maanger to Specify all WBS Items” option is selected, the restriction of only team leaders can specifiy the WBS items assigned to their team is canceled, Otherwise, the explained restriction is actived.

The integration with the MS Project 2000 may be improved to transfer the WBS items in a tree structure. Also two way integration- in addition to exporting DSPPtool data into MS Project 2000, importing MS Project data into DSPPTool- may be performed.

Although in DSPPTool, MS Access is used with a distributed support, it may not handle excessive number of users. Therefore database package of the tool can be re-implemented for some other popular database systems, ending up with a real platform independent tool.

REFERENCES

1. The Standish Group International, INC. “Chaos: A Recipe for Success”, 1999.

2. Dufner D., Kwon O., Doty A., “Improving Software Development Project Team Performance: A Web-based Expert Support System for Project Control”, IEEE Proceedings of the 32th Hawaii International Conference on System Sciences, 0-7695-0001-3/99 (c), 1999

3. Robert D. W., “Creating an Environment For Project Success”, Information Systems Management, Winter, pp. 73-77, 1997.”

4. Sommerville, I., “Software Engineering”, 5th ed. New York:Addison-Wesley Publishing Company,Inc., 1995

5. Smith, R. K., Hale, J., Parrish, A., “An Empirical Study Using Task Assignment Patterns to improve the accuracy of Software Effort Estimation”, IEEE Transactions on Software Engineering Vol. 27, No. 3, March 2001.

6. Meade, P. T., “Supporting Payload Processing With Distributed Teams”, Florida Logistics Workshop, Orlando, Florida, 2001

7. Evaristo, J. R., Scudder, R., “Geographically Distributed Project Teams: A Dimensional Analysis”, IEEE, 0-7695-0493-0/00 (c), 2000

8. Jonsson, N., Novosel, D., Lillieskold, J., Eriksson, M., “Succesful Management of Complex, Multinational R&D Projects”, IEEE Proceedings of the 34th Hawaii International Conference on System Sciences, 0-7695-0981-9/01, 2001

9. Bourgault, M., Lefebvre, E., Lefebvre, L. A., Pellerin, R., Elia, E., “Discussion of Metrics for Distributed Project Management: Preliminary

Findings", IEEE Proceedings of the 35th Hawaii International Conference on System Sciences 0-7695-1435-9/02 (c), 2002

10. Pressman, R. S., "Software engineering:A Practitioner's Approach", 3rd ed., New York: Mc Graw-Hill Inc., 1992

11. Weinrich, H., "Management: Science, Theory, and Practice", In Software Engineering Project Management. 2nd ed., R. H. Thayer, ed., IEEE Computer Society Press, Los Alamitos, Calif., 1997.

12. Project Management Institute, (PMBOK Guide 2000 Edition), "A Guide to the Project Management Body of Knowledge", Newtown Square, Pennsylvania USA., 2000

13. Fairly, R. E., Thayer, R. H., "Work Breakdown Structures", Colorado Technical University, California State University, pp. 183-194.

14. Lam, H. E., Maheshwari, P., "Task and Team Management in the Distributed Software Project Management Tool", IEEE Proceedings of the 25th Annual International Computer Software and Applications Conference (COMPSAC'01), 0730-3157/02, 2002

15. Karagöz, N. A., "Improving Individual Software Engineering skills: A Tool To Support Personal Software Process", Thesis Submitted to the Graduate School of Informatics of METU, Ankara, September 2001.

16. YAĞIZ, S., "Technical Report: Extended Version of Analysis And Design of Distributed Software Project Planning Tool (DSPPTool)", METU/II-TR-2004

17. ÜLKÜ, N., "A Distributed Software Project Management Tool", Thesis Submitted to the Graduate School of Natural and Applied Science of Dokuz Eylül University, İzmir, February 1998