Introduction to Project Management Software

Amelio Vazquez-Reina

October 19 - October 26, 2007

Abstract

In this third lab session for COMP180 we will learn how to use a project managament application. For simplicity we will stick to Microsoft Project 2003. The lab has two parts. In the first part, we will learn the basics about the Microsoft Project's environment. In the second we will create the full Gantt chart of a small-size software project.

This document is based on Microsoft Project 2003®'s official documentation and Microsoft Office Online® website's contents. This document was created for education porposes only, and the author refers the readers to the original sources. Microsoft Project is a registered trademark of Microsoft Corporation in the United States and other countries.

Contents

1 Introduction to Project Managament Tools

Project managament tools help managers to use a computer in their planning and scheduling of projects. With these tools, a project manager can define tasks, resources, budgets, Grant Charts, Pert Charts, adjust and track project costs, etc. in an easy and automated way.

A list of popular management software can be found in Wikipedia[1]. Since Microsoft Project 2003 (MP) is installed in the Tufts CS computers, we will use the Microsoft's tool for this lab.

2 Gantt chart

A Gantt chart is a project control technique that can be used for project scheduling, budgeting and resource planning [1]. A Gantt chart is a kind of a bar chart where each bar represents an activity. The bars are drawn against the timeline and the length of each bar is proportional to the length of time planned for the activity.

An example of a simple Gant Chartt can be seen in figure 1

3 Short Tutorial on Microsoft Project

3.1 Definition of preliminary concepts

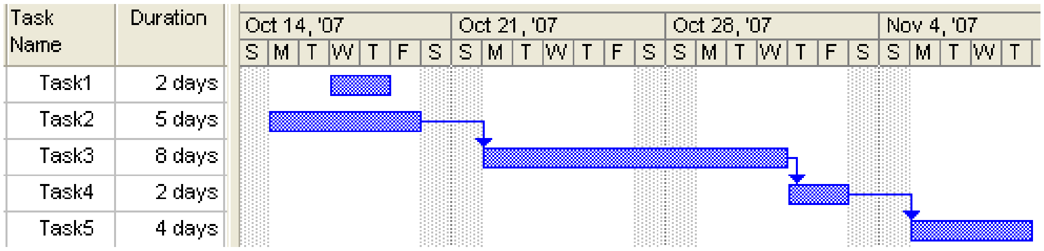
In this section we define some important concepts that will be used throught this tutorial of MP:

Tasks: A Task is an activity that has a beginning and an end. Project plans are made up of tasks.

Resources: Reasources can be people, equipment, and material that are used to complete tasks in a project.

Views: Views display, in a particular format, a subset of the information you enter in MP. That subset of information gets stored in Project and can be displayed in any view that calls for it. They appear in several formats, some of them are:

[1] `http://en.wikipedia.org/wiki/List_of_project_management_software`



Task Name	Duration
Task1	2 days
Task2	5 days
Task3	8 days
Task4	2 days
Task5	4 days

Figure 1: An MP's Gantt chart example with five tasks

Graphs: These are views that present schedule information graphically. E.g. the *Resource Graph view* graphically displays information about the allocation, work, or cost of resources in a project over time.

Charts: These are views that present project information graphically. For example, the *Gantt chart view* consists of a sheet and a chart pane where tasks are represented as horizontal bars.

Network diagrams: These are diagrams that show dependencies between project tasks. Tasks are represented by boxes, or nodes, and task dependencies are represented by lines that connect the boxes.

MP groups views in three categories: Task views, resource views and combination views. A combination view contains two views.

You can alternate between different views in the *View* menu bar.

Calendars A *calendar* is the scheduling mechanism that determines *working time* for resources and tasks. Working time defines the hours during which work can occur.

MP uses the following types of calendars:

The base calendar: A calendar that is used as the template on which the project calendar, resource calendars or task calendars are based. There are three possible base calendars: Standard (8:00 A.M. to 5:00 P.M. weekdays, with an hour off for lunch) 24-Hours, and Night Shift.

The project calendar: This is the calendar used to designate the default work schedule for all tasks in a project.

Resource calendar a.k.a. *resource working time calendar*: For each resource you enter, MP creates individual resource calendars (*working time* of each resource).

The task calendar: Tasks are scheduled based on the working times in the project calendar; however, you can customize the working times from the project calendar in a task calendar if you have tasks to be completed at different times, especially tasks that are independent of resources. A task calendar is especially useful for equipment that runs and completes tasks throughout nights or weekends, which is designated as nonworking time in the project calendar.

MP calculates when a resource and task are scheduled to work by using the calendars in the following order: 1) project calendar, 2) resource calendar, and 3) task calendar. If a task has no resources and no task calendar, it's scheduled according to the project calendar. If a task has resources assigned and no task calendar, it's scheduled according to the resource calendars.

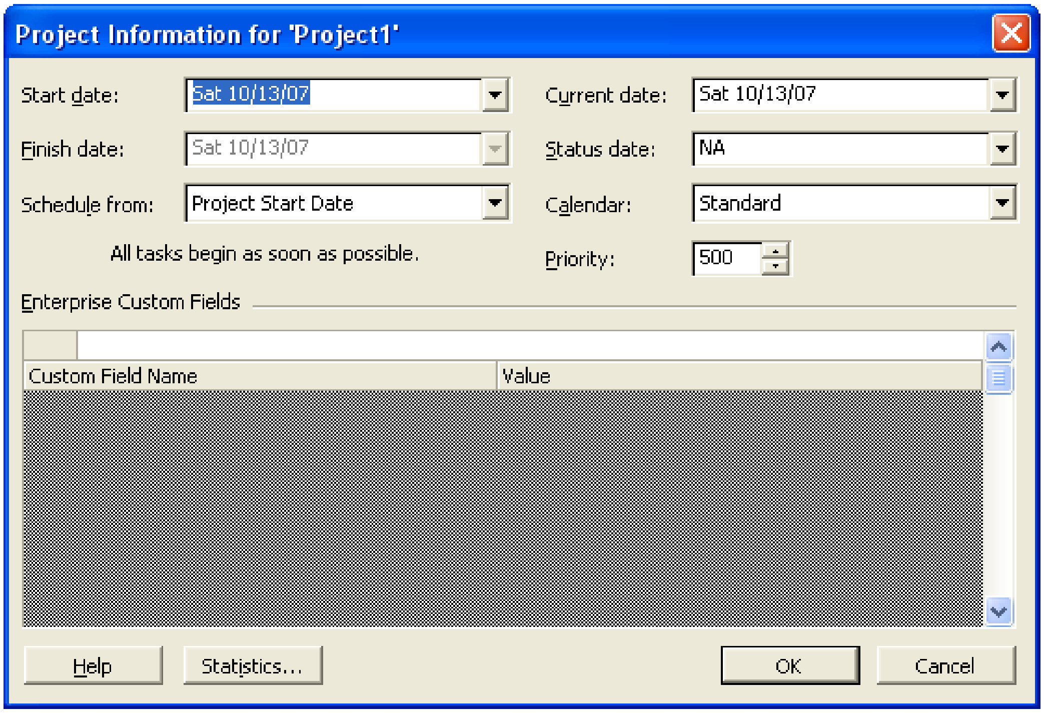
Finally, you can set the task calendar to ignore the resource calendars if you need to schedule a task during a nonworking time for a resource.

3.2 Creating a new project

Start MP, and create a new project. The first window that comes up is the *Gantt chart view*. The *Gantt chart view* shows the Gantt chart of our current project (see section 2 for more info about Gantt charts). Since we just created a new project, and we haven't defined any tasks, the Gantt chart is empty.

3.3 Setting up the project

Click on *Project* ➔*Project information*. A window like the one in figure 2 comes up. In this window you can define when the project is going to be started and which *base calendar* is to be used as the *project calendar*.



Project Information for 'Project1'

Start date: Sat 10/13/07

Finish date: Sat 10/13/07

Schedule from: Project Start Date

All tasks begin as soon as possible.

Current date: Sat 10/13/07

Status date: NA

Calendar: Standard

Priority: 500

Enterprise Custom Fields

Custom Field Name	Value

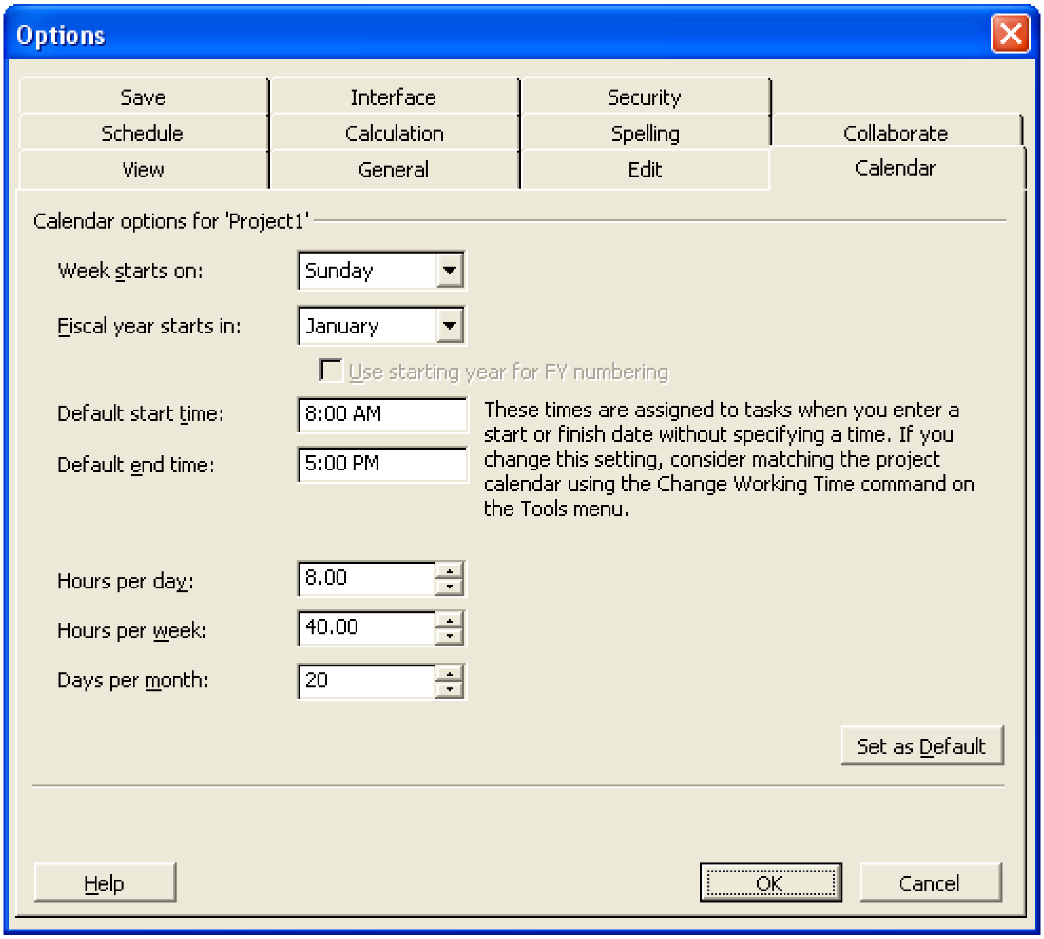
Figure 2: Setting up a project

3.4 Adjusting time and date settings

Click on *Tools* ➔*Options* ➔*Calendar*. A window like the one shown in figure 3 comes up. Here we can change default date and time settings in MP. We can specify when the project's week and fiscal year begins, the default start and end time for tasks, specify how many hours are in a day or a week, etc.

☞**Important note**: Changing options on the *Calendar tab* does not affect the *project calendar* or *resource working times* calendars. It only affects how MP converts the durations into related time amounts used throughout your project. For example, if the *Hours per day* box is set in the *Calendar tab* to the default of 8 hours, and then you enter `2d` (two days) in the *Duration* field for a task, the two days of duration is displayed on the Gantt bars as 16 hours. If you set *Hours per day* to 5, and then enter a two-day duration for a task, the task has a duration of 10 hours.

To control the way the work is actually scheduled, change the appropriate **working times** calendars. Also, if you change the number of hours per day,



Calendar options for 'Project1'

Week starts on: Sunday

Fiscal year starts in: January

□ Use starting year for FY numbering

Default start time: 8:00 AM

Default end time: 5:00 PM

These times are assigned to tasks when you enter a start or finish date without specifying a time. If you change this setting, consider matching the project calendar using the Change Working Time command on the Tools menu.

Hours per day: 8.00

Hours per week: 40.00

Days per month: 20

Figure 3: Adjusting time and date settings

hours per week, or days per month, you might want to update the project calendar to match so that times and durations are properly synchronized. To change the working times click on *Tools* ➔*Change Working Time.*

Also, settings on the *Calendar tab* apply only to new tasks entered after you change the setting. For existing tasks, the amount of work stays the same, but the duration is recalculated to use any new values for the existing amount of work.

3.5 Adding tasks to a project

Click on *View* ➔*Gantt chart*. Select the first row and type the task name. Then select the next row and repeat the process until you have entered all of the work needed to complete a part of your project. When you enter new task names, MP automatically assigns them an initial duration of one day and schedules them to start at the project start date. The task duration is the total span of active working time that is required to complete a task. This is generally the amount of working time from the start to finish of a task, as defined by the *project* and *resource calendar*.

After you enter tasks in the task list, organize and add structure to your project by applying *outlining*. *Outlining* refers to the process through which you can hide or show tasks, or show the relationship between tasks (*summary tasks* and *subtasks*). You can create subtasks by indenting a task. When you indent a task, it becomes a subtask of the nearest preceding task at a higher outline level. Subtasks share some characteristics or will be completed in the same time frame under a summary task. A summary task is a task that is made up of subtasks and summarizes those subtasks.

MP automatically determines summary task information (such as duration and cost) by using information from the subtasks. You can use summary tasks to show the major phases and subphases in the project. Summary tasks summarize the data of their subtasks. You can indent tasks as many levels as you need to reflect the organization of your project.

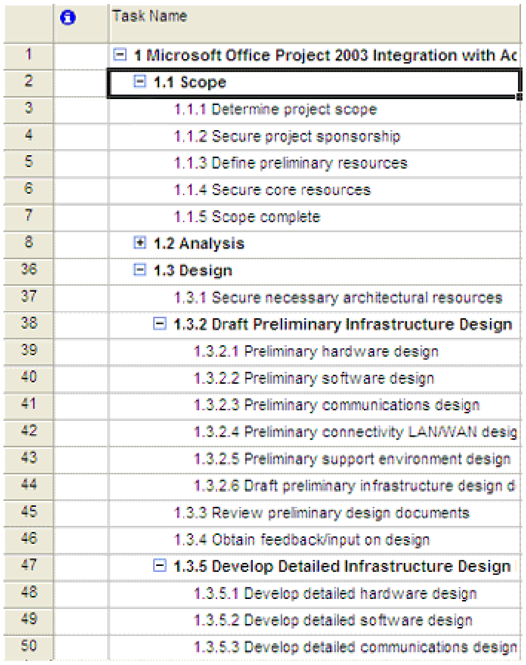
These are some task properties that you can set:

Cost: The total scheduled cost for a task, resource, or assignment, or for an entire project. This is sometimes referred to as the current cost. In Project, baseline costs are usually referred to as "budget.")

Assignment: A specific resource that is assigned to a particular task.

Work: The total labor required to complete a task. Work is different from task duration.

Something important to remember is that you can specify the time that you estimate it will take to complete the tasks by entering either *work* or *duration*. Work is the amount of effort or person hours needed to complete a task. Duration is the amount of actual time that will pass before the task is completed. Thus, if a task takes 16 hours of work and one person does the work, its duration is two days (assuming an 8-hour work day). If two people



	ℹ	Task Name
1		⊟ **1 Microsoft Office Project 2003 Integration with Ac**
2		⊟ **1.1 Scope**
3		1.1.1 Determine project scope
4		1.1.2 Secure project sponsorship
5		1.1.3 Define preliminary resources
6		1.1.4 Secure core resources
7		1.1.5 Scope complete
8		⊞ **1.2 Analysis**
36		⊟ **1.3 Design**
37		1.3.1 Secure necessary architectural resources
38		⊟ **1.3.2 Draft Preliminary Infrastructure Design**
39		1.3.2.1 Preliminary hardware design
40		1.3.2.2 Preliminary software design
41		1.3.2.3 Preliminary communications design
42		1.3.2.4 Preliminary connectivity LAN/WAN desig
43		1.3.2.5 Preliminary support environment design
44		1.3.2.6 Draft preliminary infrastructure design d
45		1.3.3 Review preliminary design documents
46		1.3.4 Obtain feedback/input on design
47		⊟ **1.3.5 Develop Detailed Infrastructure Design**
48		1.3.5.1 Develop detailed hardware design
49		1.3.5.2 Develop detailed software design
50		1.3.5.3 Develop detailed communications design

Figure 4: Summary tasks and subtasks

do the work, its duration is one day. However, **the amount of work is the same either way**.

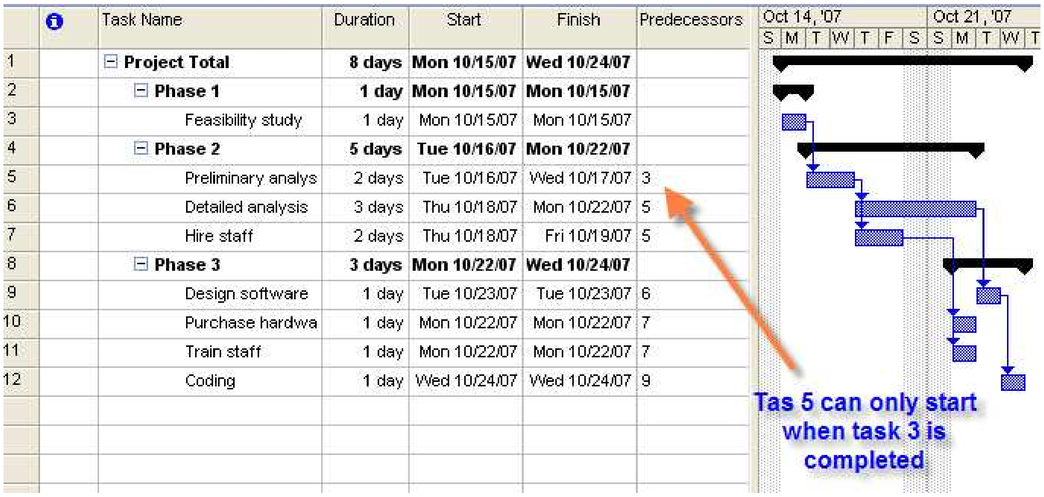
To enter work for a task, click the *Start* column heading in the table portion of the Gantt chart. On the *Insert menu*, click *Column*. Under *Field name*, select *Work*, and then click *OK*. Type work estimates in the Work column for each task, just as you would type durations.

An example with summary tasks and subtasks is shown in figure 4.

3.6 Creating dependencies between tasks

A dependency occurs when the start or finish of one task depends upon the start or finish of another. Most tasks are dependent upon other tasks. After the dependencies are set, you can easily identify the *critical path*[2] and understand the driving factors for the project end date. You can also easily make changes to one task and immediately see the ripple effect it will have

[2] A critical path is composed by the series of tasks that must be completed on schedule for a project to finish on schedule. Each task on the critical path is a critical task.



	ℹ	Task Name	Duration	Start	Finish	Predecessors
1		⊟ **Project Total**	**8 days**	**Mon 10/15/07**	**Wed 10/24/07**	
2		⊟ **Phase 1**	**1 day**	**Mon 10/15/07**	**Mon 10/15/07**	
3		Feasibility study	1 day	Mon 10/15/07	Mon 10/15/07	
4		⊟ **Phase 2**	**5 days**	**Tue 10/16/07**	**Mon 10/22/07**	
5		Preliminary analys	2 days	Tue 10/16/07	Wed 10/17/07	3
6		Detailed analysis	3 days	Thu 10/18/07	Mon 10/22/07	5
7		Hire staff	2 days	Thu 10/18/07	Fri 10/19/07	5
8		⊟ **Phase 3**	**3 days**	**Mon 10/22/07**	**Wed 10/24/07**	
9		Design software	1 day	Tue 10/23/07	Tue 10/23/07	6
10		Purchase hardwa	1 day	Mon 10/22/07	Mon 10/22/07	7
11		Train staff	1 day	Mon 10/22/07	Mon 10/22/07	7
12		Coding	1 day	Wed 10/24/07	Wed 10/24/07	9

Tas 5 can only start when task 3 is completed

Figure 5: Creating task dependencies

on the rest of the plan.

The challenge planners have is to **ensure that all tasks are in the dependency chain**. Here is a good rule of thumb: every task should have a predecessor unless it is driven by the start date of the project. Every task should have a successor unless it is the last task or milestone in the project.

When linking tasks, you can specify different types of dependencies. The most common dependency is *Finish-to-Start* (FS), which means that the predecessor task must finish before the successor task can start.

The easiest way to create task links is to enter task *IDs* in the *Predecessors* field. An example can be seen in figure 5.

To visualize the *critical path* of your Gantt chart, click on *View* ➔ *Tracking Gantt*. The critical path is visualized in red.

3.7 Adding resources to tasks

You will need people to accomplish all those tasks that you've identified in the previous section. Those people are resources that you have to manage well in order to achieve the project's objectives. You may also need some material resources - equipment, supplies, specialized environments - that you will have to schedule and pay for. You can easily include the management of resources in MS Project along with the tasks because, after all, the resources are essential to the accomplishment of the tasks in the first place. You must start by identifying the resources available along with their costs. Resource costs will be multiplied by duration to calculate project costs.

Click on *View* ➔*Resource Sheet* to specify the projet resources and costs. For each resource, enter data in each field (max units, standard rate, Ovt. rate etc.). Use the Microsoft Project 2003's documentation for information on each of resource fields. Hover the mouse over each field name to get help.

A resource can be a single person, a generic resource (generic resources: Placeholder resources that are used to specify the skills required for a specific task.) that can be replaced later, a group (for example, programmers), a piece of equipment (for example, a crane or computer), or material resources (material resources: Consumable materials or supplies, such as concrete, wood, or nails.) consumed in the course of accomplishing the task.

Once you have added all resources you have to assign them to each task. Click on *View* ➔ *Gantt chart*. In the *Task Name* field, select the task to which you want to assign a resource. Click on *Tools* ➔ *Assign resources* and choose the resources that the task is going to need.

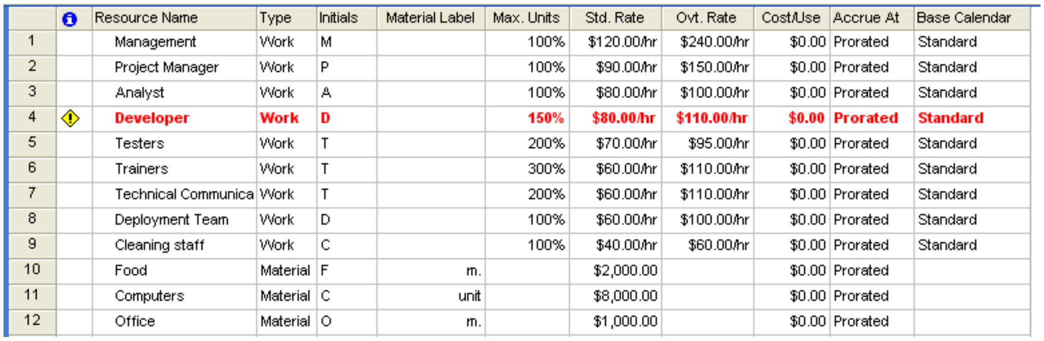
You can review how efficiently your resources are being used in your project and whether you need to make any adjustments. Some resources might be overallocated and other resources might be underallocated. Overallocation occurs as result of assigning more tasks to a resource than the resource can accomplish in the working time available. Underallocation occurs when you assign a resource to work fewer hours than the resource has available. If a resource is overallocated, MP will display it in red in the resource sheet.

Figure 6 shows a fraction of a *resource sheet*. In this example the resource `developers` is overallocated (you can either hire more developers or make them work overnight.

Resource leveling is the process of resolving resource conflicts or overallocations by delaying or splitting certain tasks. When Project levels a resource, its selected assignments are distributed and rescheduled. Generally, resources can be leveled in two ways:

- By delaying a task until the assigned resource has time to work on it.
- By splitting a task so that part of a task is done when planned and the rest of it is done later when the assigned resource has time.

  You can delay or split tasks yourself, or you can have MP do it for you, using the *Resource Leveling* feature.



	ℹ	Resource Name	Type	Initials	Material Label	Max. Units	Std. Rate	Ovt. Rate	Cost/Use	Accrue At	Base Calendar
1		Management	Work	M		100%	$120.00/hr	$240.00/hr	$0.00	Prorated	Standard
2		Project Manager	Work	P		100%	$90.00/hr	$150.00/hr	$0.00	Prorated	Standard
3		Analyst	Work	A		100%	$80.00/hr	$100.00/hr	$0.00	Prorated	Standard
4	⚠	**Developer**	**Work**	**D**		**150%**	**$80.00/hr**	**$110.00/hr**	**$0.00**	**Prorated**	**Standard**
5		Testers	Work	T		200%	$70.00/hr	$95.00/hr	$0.00	Prorated	Standard
6		Trainers	Work	T		300%	$60.00/hr	$110.00/hr	$0.00	Prorated	Standard
7		Technical Communica	Work	T		200%	$60.00/hr	$110.00/hr	$0.00	Prorated	Standard
8		Deployment Team	Work	D		100%	$60.00/hr	$100.00/hr	$0.00	Prorated	Standard
9		Cleaning staff	Work	C		100%	$40.00/hr	$60.00/hr	$0.00	Prorated	Standard
10		Food	Material	F	m.		$2,000.00		$0.00	Prorated	
11		Computers	Material	C	unit		$8,000.00		$0.00	Prorated	
12		Office	Material	O	m.		$1,000.00		$0.00	Prorated	

Figure 6: A small fraction of a resource sheet where overallocation occurs.

4 Managing a small-size software project

We are ready to start managing a small software project. Imagine you are the project manager of a company with ten employees. Think of a small size software project (designing a website, developing a casual video-game, etc.). Your work is to design the full Gantt chart of the project, setting up budgets, hardware resources, task durations, task dependencies, and assigning labor work to each task.

You have complete freedom to manage the project as you'd like. You can virtually set any reasonable budget for the project and define the task dependencies as you want. The only restriction that you have is that you should complete the project in less than one year.

This is a guideline of the main phases/tasks that your project could have:

Scope phase:. Determining project scope, defining preliminary resources, etc.

Analysis phase:. Conducting needs analysis, drafting preliminary software spefications, developing delivery timeline, etc.

Design phase: Developing functional specifications, developing prototypes, incorporating feedback on the design, etc.

Development phase: Assigning development staff, developing code, primary debugging, etc.

Testing phase: Developing unit test plans using product specifications, developing integration test plans, performing unit testing and integration testing, etc.

Documentation phase: Developing the help system, incorporate help documentation feedback, developing user manuals, etc.

Pilot phase: Identifying a test group, installing/deploying software, obtaining user feedback, evaluating testing information, etc.

Deployment phase: Developing deployment methodology, training support staff, deploy software, etc.

Post Implementation Review: Documenting learned lessons, creating software maintenance team, etc.

5 What to hand in

Following the indications in section 4, create a `.mpp` file with the full Gantt chart of your project. Submit the `.mpp` file along with a `.tex` file that uses the ACM class and which includes:

- Your comments about the critical path of your Grantt chart. Try to identify potential problems that might arise in the execution of your plan.

- Relation between your plan and the software process models learned in class. E.g. How is it your plan similar to the water-fall model, and iterative/incremental/evolutionary models?

- Your thoughts about using a project management such as Microsoft Project. What is it most useful for? What information is missing in the Gantt chart?

This lab will span two Friday labs (October 19 and October 26). Zip all files and submit your lab through Moodle before **Friday October 26 at 23:59**.

References

[1] Carlo Ghezzi, Mehdi Jazayeri, and Dino Mandrioli. *Fundamentals of Software Engineering*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2002.