[Team Name]

[Project Name]

Version #

Date

Powered By
www.construx.com

Portions Copyright © 2001 Construx Software Builders, Inc.

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Draft Type and Number	Full Name	Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded.	00/00/00

The paragraphs written in the "Comment" style are for the benefit of the person writing the document and should be removed before the document is finalized.

A project plan is the controlling document for managing a software project.

This project plan template is intended to scale with the size of a project. On very small projects, the project plan may be the only documentation for a project. On very large projects, a family of project plans may exist. Most projects will lie somewhere in-between, with the project plan being a combination of direct information and pointers to other project documents.

Review & Approval

If appropriate, this section contains formal sign-off for both review and approval of the project plans. Normally the projects authority and agent should formally sign off the plan to launch a project. If you do not need all this approval, you can delete this section.

Project Plan Approval History

Approving Party	Version Approved	Signature	Date
Authority 1			
Authority n			
Agent			

Project Plan Review History

Reviewer	Version Reviewed	Signature	Date
Authority 1			
Authority n			
Agent			
Stakeholder 1			
Stakeholder n			
Etc.			

1 Introduction

The introduction section provides context for the project.

1.1 Overview (Executive Summary)

This section normally consists of a summary of project purpose, scope, and objective. It may also include background for the project, a summary of the project planning, or other contextual material as appropriate. This should be a nice high-level description of the project so that executives, who do not want to read the entire document, still have a good idea what you are going to do. It is important, because once you get a good, concise, well-written summary of the project, you can reuse this section in the overviews of the SRS (Software Requirements Specification) and SDS (Software Design Specification).

1.2 Deliverables

What are the ultimate deliverables that will be delivered to the customers or consumers of the project? Everything you need to produce should be listed here. Do not just put Delivery Documentation or Software. What documents exactly make up the delivery documentation? A tutorial or just a user's manual? What makes up the Software delivery? Source code, or code and executable and install scripts, etc. This becomes your "contract" of everything tangible you have to produce on the project. This way you are protected for someone showing up the last week of class and saying, "Oh, didn't you know, I wanted a full auto-install program also."

1.3 Assumptions and Constraints

A formal or informal collection of assumptions, constraints, external dependencies, etc. These are not technical constraints per se on the system you are building, there are constraints that affect the execution and planning of the project. Project Schedule must be 15 weeks or less. Man-Hours must be 300 hours or less. Cannot use certain libraries or languages. Must use certain libraries and languages. All this kind of stuff affects the schedule and the scope of the work you can complete in the semester.

1.4 Reference Materials

Bibliographic listing of non-project materials referenced in the plan..

1.5 Definitions and Acronyms

List any project definitions and acronyms introduced to the project by this plan. Do not repeat items covered in a global project definitions and acronyms document. Be sure and define all acronyms before their use. This section can establish a common project vocabulary so everyone on the project understands technical terms that will be used. These are typically specialized domain terms that normal project members or customers would not *be familiar with, but which are key to the planning document.*

2 Management Structure

This section covers all aspects of managing the project that are not related to estimating, planning, and controlling the work.

2.1 Project Lifecycle

Describe in detail the lifecycle for the project. Normally this section discusses phases, deliverables, and dates abstractly to provide a clear view of the project time line from a process standpoint and to avoid duplicating content from other sections. You may reference a standard lifecycle (like the waterfall) and then note how you will deviate from that.

2.2 Project Organization

Describe the organization structure (management, groups, roles, etc.) of the project and relationships to the external organization or other projects.

2.2.1 External Interfaces

This section is not for external interfaces to the application you are developing. Rather it identifies who outside the project team has an important role in this project. For instance a faculty advisor or customer is a key external interfaces for your project team. What is the process for contacting them? Can anyone call them up, only the team leader or a designated point of contact?

2.2.2 Internal Structure

2.2.2.1 Roles and Responsibilities

A table like the one below can be used, roles shown are illustrative of a software project. You may decide to do different roles.

Role	Responsibility
Project Manager	
Planning and Tracking Lead	
Requirements Lead	
Quality Assurance Lead	
Design Lead	
Implementation Lead	
Development Engineer	
QA Engineer	

2.2.2.2 Staffing

Contains the schedule of assignment of roles and responsibilities to particular individuals.

Role	Staff Member	Start Date	End Date
Project Manager			
Planning and Tracking Lead			
Requirements Lead			
Quality Assurance Lead			
Design Lead			
Implementation Lead			
Development Engineer 1			
Development Engineer n			
QA Engineer 1			

2.3 Communication

Describe the types of communication that need to occur during the project and their timing. What is your meeting schedule? How will you maintain contact?

2.4 Risk and Asset Management

Describe how risks and assets will be actively managed on a project. Risk identification and mitigation strategies go here. Typically, managers on real projects closely monitor their top ten risks. For this project, you might want to focus on your top 5. Risks are not just technical, they may deal with people/team related issues, changing requirements, spending too much time in the "fuzzy front end" or risky algorithms. A good generic list is off the sample risks link on the class web page.

2.5 Startup

What special steps are necessary to get the project up and running?

2.6 Closeout

How will the project ramp down and end or transition into a new project?

3 Planning and Control

This section covers all aspects of managing the project related to estimating, planning, and controlling the work. Most large project plans will contain pointers to specialized artifacts and group of artifacts for the items in this section.

3.1 Estimate

Describe any and all estimates to date relating to the project. How did you arrive at your estimate of time and code size? What is the initial estimate? Does this map to the time available for this semester?

3.1.1 Estimation Process

How and when will estimates be refined during the project? How will you track and refine your estimates to be sure you are zeroing in on an accurate overall estimate for the project.

3.2 Resource Identification

List the resources available to the project including information on available staff, time, cost, materials, etc. When appropriate, describe the resource obtainment and phase out timelines.

3.2.1 Staff

Unlike the staffing section that assigns individuals to the project, this section deals with abstracted planning of the peak staff, staffing profile, and staff build-up of the project.

3.2.2 Time

What is the available calendar time for the project. This section normally consists of detailed constraints. Obviously for this class, you have a semester constraint. If the customer wants an earlier delivery, then that shortened time would be documented here.

3.2.3 Cost

The available budget for the project. You have no financial budget. In this class we track everything by man-hours. The total "budget" is a function of the number of team members that you have. A credit-hour represents a minimum of 50-60 hours of work. Therefore for this class, each team member represents 150-180 hours of work. For a nominal four-person team, your available budget would be a minimum of 720 hours.

3.2.4 Materials

Materials that will be used for the project, if any. Do you need specialized equipment for the project? How will you get that? Note, if uncertain how you will get required equipment, then this should also be a risk that has been identified for the project.

3.3 Resource Allocation

Describe the allocation of the resources to meet the project goals. This section will at a minimum refer to a WBS (Work Breakdown Structure) and top-level project schedule. It may also refer to multiple schedule views, budgeting plans, etc.

3.3.1 Milestones

This section lists key milestones for the projects. Milestones should have a name and a due date. Milestones are NOT tasks. Tasks are covered in section 3.3.2. Milestones consume no resources or time, they are like binary yes/no checkpoints for the project. Ideally, you should have a milestone a week for the project. That way you know you are never more than a week behind. Some milestones are given to you, like Design Document Turned-In. Others you will need to develop, like Initial Prototype Tested. Status Report Due is not a good milestone. You can hit all your status reports and still fail the project. You need to pick milestones that will help you track your project status.

3.3.2 Work Breakdown Structure

This section refers to the WBS for the project, which is normally an external document in large projects.The WBS breaks down all work on the project into work packages or tasks. You should account for all activities here that will eat up your budget hours in paragraph 3.2.3. You should break down the tasks to a level so that you can assign them to a specific person, you understand the task well-enough to do it, and you can determine when you are done (the exit criteria).

3.3.3 Schedule

This section refers to schedules for the project. There are usually several different schedules of varying levels of detail that describe the allocation of work packages. Remember that a schedule = tasks (from 3.3.1), a sequence (what tasks are done when and which are prerequisites for other tasks), an allocation of time (your hours budget) and resources (one of your team members), and an integrated set of milestones. The best way to show a schedule is with an annotated Gantt or CPM chart. Most students use Gantt charts, because that is what is supported in MS Project, which many student teams use. Visio supports both Gantt and PERT charts.

3.4 Tracking and Control

Describe how project cost, schedule, quality, and functionality will be tracked and controlled throughout the project. How do your teammates report their individual accomplishments for inclusion in the weekly status report? Who is responsible for tracking the number of hours expended on tasks?

4 Technical Process

Describes or summarizes the top-level technical processes used on the project.

4.1 Engineering

Covers environment, tools, methods, and techniques related to software engineering.

4.1.1 Environment

4.1.2 Methods, Tools and Techniques

What CASE tools (like Rational Rose or Visio) will you use? Will you use UML or another modeling technique? Will everyone prepare documents in Word, in text files, or in PDF files? Will everyone communicate with email?

4.2 Technology

Covers environment, tools, methods, and techniques related to technology.

4.2.1 Environment

4.2.2 Methods, Tools, and Techniques

What IDE will you use? What type of computers? What other technology will the project standardize on to accomplish the project?

4.3 Infrastructure

What needs to be done to create development environments?

4.4 Project Artifacts

If appropriate, this section describes all planned artifacts or groups of artifacts for a project. If this is already in deliverables section then you can omit here.

5 Supporting Plans

This section either incorporates or references via pointer detailed supporting plans.

5.1 Configuration Management

Identification and organizational of project artifacts, change control, builds, and release processes. Often the different components of configuration management may be broken into separate specialized plans. What method or tool will you use on your project (like CVS)?

5.2 Quality Assurance

Planning, tracking, and control for ensuring project meets quality goals. Also discusses technical process specific to quality assurance, such as reviews. How will your team track and report bugs.How will be sure everyone is following the plan?

5.3 Testing

Planning for execution and tracking of test activities, including test coverage planning Are you going to do any project-standard testing (like using JUnit, etc.)?

5.4 Deployment

Normally this section will point to a dedicated document if there are any significant deployment issues. How would you hand off this project to your customer.

5.5 Integration

Covers integration between sub-projects, tasks, releases, or other significant project integration.

5.6 Procurement

Covers significant procurement of materials or sub-contracted services on a project.

5.7 Operations

Covers issues related to system operation after deployment. Normally this section will point to a dedicated document if it is needed. This may refer to the installation guide which will be written later in the project.

5.8 Maintenance

Covers issues related to maintenance phase of a project. If the project itself is a maintenance effort this would not be needed.

5.9 Staff Development

Covers significant training, teambuilding, or other staff development activities, e.g., a hiring plan. For this class, be sure and account here for hours that need to be devoted to learning domain, doing research and learning new languages or development environments.

5.10 Product Acceptance

If significant customer acceptance activities are necessary that cannot be handled as part of the test plan.