**Revision Chart**

*This chart contains a history of this document’s revisions. The entries below are provided solely for purposes of illustration. Entries should be deleted until the revision they refer to has actually been created.*

*The document itself should be stored in revision control, and a brief description of each version should be entered in the revision control system. That brief description can be repeated in this section. Revisions do not need to be described elsewhere in the document except inasmuch as they explain the development plan itself.*   
  

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| Draft | TBD | Initial draft created for distribution and review comments | TBD |
| Preliminary | TBD | Second draft incorporating initial review comments, distributed for final review | TBD |
| Final | TBD | First complete draft, which is placed under change control | TBD |
| Revision 1 | TBD | Revised draft, revised according to the change control process and maintained under change control | TBD |
| Revision 2 | TBD | Revised draft, revised according to the change control process and maintained under change control | TBD |
| etc. | TBD | TBD | TBD |

**Contents**

*List the document contents by page number.*

**List of Figures**

*List the figures in the document by page number.*

**1.0 Introduction**

*This section should describe the project and the software product being to be built. No text is necessary between the heading above and the heading below unless otherwise desired.*

**1.1 Project Overview**

*Give a short summary of the project objectives, the software to be delivered, major activities, major deliverables, major milestones, required resources, and top-level schedule and budget. Describe the relationship of this project to other projects, if appropriate.*

**1.2 Project Deliverables**

* 1. Submit a project report to answer the following questions?

How many members are you in your team? List all team members.

What type of team model is used in the project (chapter 4)?

Paste all UML diagrams you used in the project. Some important diagram are expected to be included like architecture diagram, use case diagram, class diagram, and so forth.

Submit all artifacts you used in the project. These artifacts include source code, UML diagrams, SPMP, version control documentation, test cases, data storage files, and other necessary artifacts we studied in the class.

**1.3 Evolution of the Software Project Management Plan**

*Describe how you expect this document to evolve over time. This section should be very similar to the "Revision Chart" earlier in the document. The revision chart should list what has already been done to this document. This section should list what is expected to be done to this document.*

*This chart contains a plan for the document’s revisions. The entries below are provided for purposes of illustration. The progression from "Draft" to "Preliminary" to "Final" and then through a section of "Revisions" is one typical approach.*   
  

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Primary Author(s)** | **Description of Version** | **Date Expected** |
| Draft | TBD | Initial draft created for distribution and review comments | TBD |
| Preliminary | TBD | Second draft incorporating initial review comments, distributed for final review | TBD |
| Final | TBD | First complete draft, which is placed under change control | TBD |
| Revision 1 | TBD | Revised draft, revised according to the change control process and maintained under change control | TBD |
| etc. | TBD | TBD | TBD |

**1.4 Reference Materials**

*List all the documents and other materials referenced in this document. This section is like the bibliography in a published book.*

**1.5 Definitions and Acronyms**

*Provide definitions or references to all the definitions of the special terms and acronyms used within this document.*

**2.0 Project Organization**

*In this section, describe the process model (e.g., lifecycle model), the organizational structure (e.g., chain of command or management reporting structure), and responsibilities of individuals on the project.*

*No text is necessary between the heading above and the heading below unless otherwise desired.*

**2.1 Process Model**

*Describe the following items:*

* *The project’s lifecycle model (e.g., waterfall model, spiral model, evolutionary prototyping model, etc.)*
* *The project’s major milestones (content and dates/timing). This should include a text description of the meaning of each milestone plus a Gantt chart or other high-level description of the project’s schedule.*
* *Major work products, including content and timing. These should be listed in a table like the following one:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Work Product Name** | **Planned Completion Date** | **Placed Under Change Control?** | **Deliverable to Customer?** | **People Who Must Sign Off on the Work Product** |
| Software Project Management Plan  This is the controlling document for managing a software project, and it defines the technical and managerial processes necessary to deliver the project requirements. | TBD | Yes | No | Project Manager, Engineering Lead, QA Lead, Documentation Lead |
| TBD | TBD | TBD | TBD | TBD |

*You should consider including all of the top-level work products described in "Software Project Survival Guide", including:*

* *Change Control Plan*
* *Change Proposals*
* *Vision statement*
* *Top 10 Risks List*
* *Software Development Plan, including project cost and schedule estimates*
* *User Interface Style Guide*
* *User Manual/Requirements Specification*
* *Quality Assurance Plan*
* *Software Architecture*
* *Software Integration Procedure*
* *Staged Delivery Plan*
* *Individual Stage Plans, including miniature milestone schedules*
* *Coding Standard*
* *Detailed Design Documents*
* *Software Construction Plans*
* *Deployment Document (Cutover Handbook)*
* *Release Checklist*
* *Release Sign-Off Form*
* *Software Project Log*
* *Software Project History Document*

*The items in this section (and, in general, in this document) can be fully described here, or, if they are described in sufficient detail elsewhere, that description can simply be referenced here.  [Try to avoid redundancy with section 4.2].*

**2.2 Organizational Structure**

*Describe the internal management structure of the project. Use org charts, matrix diagrams, or other appropriate notations to describe the lines of authority, responsibility, and communication within the project.*   
*[It is NOT necessary to show the organization of the entire company, only this project].*   
*[This section lists project roles or jobs, but does not mention who is assigned to them.  Actual personnel assignments are shown in section 3.5].*   
*[Refer to*[*Job Descriptions*](http://www.csc.calpoly.edu/~jdalbey/205/Mgmt/jobs.html)*].*

**2.3 Organizational Boundaries and Interfaces**

*Describe the relationships between the project and each of the following organizations:*

* *Parent organization (upper management)*
* *Customer organization (internal or external)*
* *Subcontracting organization(s) (if any)*
* *QA organization, if separate*
* *Documentation organization, if separate*
* *End-user support organization, if separate*
* *Any other organizations the project interacts with*

*This list should include a description of a specific person or project role that is responsible for maintaining the interface between the project and each of these other organizations.*

*[In CPE 205 the two important interfaces are Analyst -> Customer and Team Leader -> Course Instructor].*

**2.4 Project Responsibilities**

*[Refer to*[*Job Descriptions*](http://www.csc.calpoly.edu/~jdalbey/205/Mgmt/jobs.html)*]. Identify and describe each major project function and activity, and identify the person(s) responsible for each function and activity. If appropriate, you might use a table like the one below, otherwise just write a narrative description.* 

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Manager** | **QA** | **Analyst** | Designer | User Interface |
| Liaison Responsibility | VP Engineering |  | Customer | Software engineering lab |  |
| Document Responsibility | SPMP | SQAP | SRS | SDD | UI Prototype |
| Other Responsibilities |  | FTR's |  |  |  |

**3.0 Managerial Process**

*Describe management objectives, priorities, project assumptions, dependencies, constraints, risk management techniques, monitoring and controlling mechanisms, and the staffing plan.*

*No text is necessary between the heading above and the heading below unless otherwise desired.*

**3.1 Management Objectives and Priorities**

*Describe the philosophy, goals, and priorities for management during the project. Consider including the following items:*

* *relative priorities among functionality, schedule, budget [e.g., student hours worked], and quality.*
* *risk management procedures*
* *approach to acquiring third party software*
* *approach to modifying or using existing software*

**3.2 Assumptions, Dependencies, and Constraints**

*Describe the assumptions upon which the project plans are based, the dependencies of the project plans, and the constraints upon the project plans (e.g., schedule, budget, quality, functionality, etc.)*

**3.3 Risk Management**

*Describe how risks will be identified, tracked and monitored. Identify the risk manager and his/her duties. The actual risks themselves are contained in a separate document, the Risks List, that will be updated weekly. Place a link here to the Risks List.*

**3.4 Monitoring and Controlling Mechanisms**

*Describe how project cost, schedule, quality, and functionality will be tracked throughout the project. Consider describing the following:*

* *Status and Progress Report contents/formats*
* *Reporting structure and frequency*
* *Meeting structure and frequency*
* *Audit mechanisms*

**3.5 Staffing Plan**

*Describe the numbers and types of personnel needed to conduct the project. Describe the required skill levels, start times, duration on the project, method of obtaining the personnel, training required, and phasing out of project personnel. [Once staff have been hired, list their names and roles here.]*

**4.0 Technical Process**

*This section describes the top-level technical processes used on the project including the technical methods, tools, and techniques; major software documents; and supporting activities such as configuration management and quality assurance.*

*No text is necessary between the heading above and the heading below unless otherwise desired.*

**4.1 Methods, Tools, and Techniques**

*Describe the following:*

* *The computing system environment including hardware and operating system environment;*
* *Software tools including design tools, source code control, time accounting, compiler or IDE, debugging aids, defect tracking, and so on*
* *Development methodologies including requirements development practices, design methodologies and notations, programming language, coding standards, documentation standards, system integration procedure, and so on (these will not all be defined when the first draft of the project plan is created; the section should be updated as the plans become more detailed)*
* *Quality assurance practices including methods of technical peer review, unit testing, stepping through code in a debugger, system testing, automated regression tests, and so on*

**4.2 Software Documentation**

*List the documents that will be developed for the project, including milestones, reviews, and signoffs for each document. If they are described in sufficient detail elsewhere, that description can simply be referenced here. The documentation list might include the follow documents:*

* *Change Control Plan*
* *Change Proposals*
* *Vision statement*
* *Top 10 Risks List*
* *Software Development Plan, including project cost and schedule estimates*
* *User Interface Style Guide*
* *User Manual/Requirements Specification*
* *Quality Assurance Plan*
* *Software Architecture*
* *Software Integration Procedure*
* *Staged Delivery Plan*
* *Individual Stage Plans, including miniature milestone schedules*
* *Coding Standard*
* *Detailed Design Documents*
* *Software Construction Plans*
* *Deployment Document (Cutover Handbook)*
* *Release Checklist*
* *Release Sign-Off Form*
* *Software Project Log*
* *Software Project History Document*

**4.3 Project Support Functions**

*Describe or give references to other documents that describe the plans for functions that support the software development effort, including configuration management, quality assurance, and end user documentation. Whether included in this document or described elsewhere, the plans for these supporting functions should be developed in as much detail as the plans for the software itself are. (This includes responsibilities, resource requirements, schedule, budget, and so on.)*

**5.0 Work Packages, Schedule, and Budget**

*No text is necessary between the heading above and the heading below unless otherwise desired.*

**5.1 Work Packages**

*Describe the work package (i.e., task or collection of tasks) that must be completed to complete the software. Identify each work package with a unique number, and provide a diagram showing the breakdown of work packages into sub-packages.*

**5.2 Dependencies**

*Describe the dependencies both among work packages and between the project’s work packages and external events.*

**5.3 Resource Requirements**

*Describe the resources required for the project week by week or month by month, depending on the length of the project. Describe the numbers and type of personnel that are used, number of computers and software used, office facilities used, training needed, budget used, and so on over the whole course of the project.*

**5.4 Budget and Resource Allocation**

*Describe the portion of the project’s budget allocated to different functions (e.g., engineering, quality assurance, documentation, management). Describe how the budget will be expended over the course of the project.*

**5.5 Schedule**

*Describe the schedule for the various project functions, activities, and tasks, taking into account the various dependencies and required milestone dates and deliverable dates. You can express schedules either as absolute calendar dates or as times relative to key project milestones (e.g., "requirements signed off + 60 days").*

*[The schedule is updated frequently and may be more appropriately maintained as a separate document, in which case include a link here. ]*

**6.0 Additional Components**

*Include additional components needed to manage your specific project. Possibilities include subcontractor management plans, security plans, training plans, hardware procurement plans, facilities plans, installation plans, cutover plans, and software maintenance plans.*

**7.0 Index**

*The index is optional according to the IEEE standard. If the document is made available in electronic form, readers can search for terms electronically.*

**8.0 Appendices**

*Include supporting detail that would be too distracting to include in the main body of the document.*

This material is copied and/or adapted from the Survival Guide Website at [www.construx.com/survivalguide/](http://www.construx.com/survivalguide/). This material is copyright © 1993-1998 Steven C. McConnell. Permission is hereby given to copy, adapt, and distribute this material as long as this notice is included on all such materials and the materials are not sold, licensed, or otherwise distributed for commercial gain.