

Doc Project Planning Outline

Companion document for "Planning a Documentation Project" (by David Wolicki)

1.0 Setting Your Goals

Decide what you want to have at the end of the planning process.

1.1 A complete list of deliverables

1.2 A schedule for what will be required when

1.3 An indication of who will do the doc work

1.4 Approval of this information and inclusion of it in the overall project plan

2.0 Researching a new Project - *The Five Ws for Docs*

Asking these questions will help you gather the information that you will need to prepare your doc plan/proposal.

2.1 What

2.1.1 Is the product similar to an existing one?

2.1.2 What are the major components?

2.1.3 What will the product be used to do?

2.1.4 What environment will the product require (Operating systems, Hardware environment etc.)

2.2 Why

2.2.1 Why would a user want to buy this product?

2.2.2 Why does the company need this product? (Does it fill a hole in the product line?)

2.3 How & Where

2.3.1 How does the product work (infrastructure)?

2.3.2 How will it be used?

2.3.3 How will it be sold/marketed?

2.3.4 How will it be developed (what architecture and tools)?

2.3.5 Where will this project be developed?

2.4 When

2.4.1 When will the design phase be complete (i.e., when will design docs be available and approved)?

2.4.2 When will development be frozen?

2.4.3 When will the product be delivered to Testing/QA/QC?

2.4.4 When will the product be released (Beta or GA)?

2.5 Who

2.5.1 Who is developing the product?

2.5.2 Who will provide info to the doc team?

2.5.3 Who will review the docs?

2.5.4 Who will use the product?

2.5.4.1 What is their knowledge level and core skill set?

2.5.4.2 What is their native language?

2.5.4.3 Is there a specific customer for this product?

3.0 Compiling a Doc Plan/Proposal

Use the information gathered in the previous phase to build a plan.

3.1 Determine a set of deliverables.

This list should fill the following requirements:

3.1.1 It should provide all information that the user will need in order to use the product.

3.1.2 It should be consistent with the product positioning (describe the product in the same way that it is being sold).

3.1.3 It should be consistent with our knowledge of the typical users (knowledge base, language level, core skills etc.,).

3.1.4 It should include all textual elements of the product that need to be reviewed/created by the documentation team (screen elements, help, etc.,).

3.1.5 It should list the general contents (chapter titles), format, and purpose of each deliverable.

3.2 Determine Resource Needs

3.2.1 Determine the number of man-hours/days/weeks needed for each deliverable.

This should be based on:

3.2.1.1 Project scope (assemble a task list if possible)

3.2.1.2 Comparison with documents for a similar project

3.2.1.3 Experience levels of the writing and development teams

For help in this area, see <http://www.comtech-serv.com/> JoAnn Hackos' site.

- 3.2.2 Determine if the current resources have time available for the deliverables that were defined above. If not, indicate that external resources may be necessary.
- 3.2.3 Determine if new tools (hardware or software) will be needed for the project.
- 3.2.4 Determine needed data resources (e.g., information about new technology or formats).

3.3 Set a schedule for each deliverable.

- 3.3.1 Indicate when external resources will be needed (e.g., reviewers).
- 3.3.2 Indicate major milestones (e.g., outline, and first handoff to QA).

3.4 Submit a Proposal

Include the following information (gathered in the previous steps):

- 3.4.1 List of Deliverables
- 3.4.2 Schedule
- 3.4.3 Needed Resources

Be sure to get your plan approved and incorporated with the overall project plan.

4.0 Where To Go From Here

Here are some ideas as to what you will need to do in the next phase of your project:

- 4.1 Prepare preliminary outlines for each deliverable.***
- 4.2 Submit outlines for approval.***
- 4.3 Ensure that no resource conflicts exist between this project and other projects on your "table".***
- 4.4 Anticipate potential bottlenecks and take preventive measures (e.g., is there something that you can get started early thus preventing a problem later?).***
- 4.5 Get information about the future of this project (maintenance schedules, new versions, etc.).***