# Requirements Specification Document

### Assignment Learning Objectives

Be able to

* Write clear, complete, and unambiguous **requirements specifications**
* Write and format a technical report
	+ Use tables with **captions** and row/column headings to effectively present information
	+ Use the **inverted pyramid style of writing** to present key information and conclusions first
	+ Write paragraphs that begin with strong topic sentences, demonstrate unity, and flow logically and smoothly from one to the next
* Prepare **software engineering diagrams** to help communicate the project requirements

## Assignment

Create a **Requirements Specification Document** containing the following sections:

1. **Title Page**
	* Include your team name & logo, project name, and date
	* Include the name(s) of the document editor and list as authors the team members who wrote content for this document and what content they wrote
		+ Do not list as authors the team members who did **not** write content
	* Include your customer and/or faculty advisor names
2. **Table of Contents**
	* Consider using Word's built-in automatic table of contents feature
	* The first page of the Introduction Section should appear as page 1
	* Do not include page numbers on the Title Page or Table of Contents page
	* Do not include the title or table of contents section in the Table of Contents
	* Include a Table of Tables and Table of Figures if appropriate
3. **Introduction**: This section provides an overview of the entire document
	* Project Overview: Briefly describe the real-world problem
	* Project Scope: Briefly mention the most important features and constraints of your program
	* Document Preview: Describe the purpose, scope, and intended audience of this document. Preview the major sections that follow.
4. **Project Overview**: This section elaborates on the topics introduced in the Introduction Section and provides background information on the factors effecting the scope of the project and its requirements
	* Begin this section with a carefully written introduction paragraph(s) previewing and summarizing the section's contents
	* Identify the customer, stakeholders, and the intended users of your system
	* Provide a complete description of the problem being solved
		+ Explain the current system or solution in place
		+ Justify your proposed solution to this problem and explain why your solution needs to be developed rather than just bought
	* Describe the main features of your proposed system
	* Mention the most important constraints that may influence design decisions (compatibility, reliability, hardware limitations, interfaces to other systems, etc.)
5. **Development and Target Environments**
	* Begin this section with a carefully written introduction paragraph(s) summarizing the section's contents and previewing the subsections that follow
	* Describe the hardware and software resources necessary to build and run the system
6. **System Model**
	* Begin this section with a carefully written introduction paragraph(s) summarizing the section's contents and previewing the subsections that follow
	* Present a high-level view showing the major components of the existing **and** proposed system and their relationships with each other
	* Use text descriptions that refer to graphical representations such as block diagrams that are included as figures in this section or as appendices
7. **User Interaction**
	* Describe the actions of your program from the point of view of the user
	* Use-case diagrams and **use case scenarios** are an effective way to describe the interaction
8. **Functional Requirements**: This should be the **largest and most important section** of the document
	* Begin this section with a carefully written introduction paragraph(s) summarizing the section's contents and previewing the subsections that follow
	* Describe in clear, unambiguous terms the **functional requirements** of the system
		+ All requirements should be uniquely identified
		+ All requirements must be specified so that they are verifiable
		+ Requirements should be prioritized to enable trade-offs
	* Provide a sufficient level of detail for designers to design a system satisfying the requirements and testers to verify that the system satisfies requirements
9. **Nonfunctional Requirements**: Detail the constraints under which your system must operate.
	* Begin this section with a carefully written introduction paragraph(s) summarizing the section's contents and previewing the subsections that follow
	* Typical **nonfunctional requirements** deal with compatibility, efficiency, reliability, portability, memory size constraints, response time, problem size, and so on
	* Describe in clear, unambiguous terms the **nonfunctional requirements** of the system
		+ All requirements should be uniquely identified
		+ All requirements must be specified so that they are verifiable
		+ Requirements should be prioritized to enable trade-offs
10. **Feasibility**
	* Begin this section with a carefully written introduction paragraph(s) summarizing the section's contents and previewing the subsections that follow
	* Make sure your project has a chance of being completed by the end of the semester
	* Sketch out two versions of your system: a bare bones version that delivers the essential features (which you are confident of finishing) and an enhanced version that incorporates all the desired features
11. **Conclusion**
	* Include a short conclusion section to signal to the reader that the document is ending
12. **Appendices**
	* System block diagrams
	* Entity-Relationship or Database Model diagrams (for projects involving databases)
	* Website maps (for Web projects)
	* Use-case diagrams
	* Others as appropriate (optional)

### Grading criteria

Your grade will be based on both your demonstrated writing proficiency and on the contents of the document.

Two scoring rubrics will be used in assessing this document: a content scoring rubric and a writing proficiency scoring rubric. You are encouraged to print these rubrics and use them as checklists for expectations, writing guidelines, and quality assurance.

**Honor code**: The work needs to be your own. You may wish to have someone from outside the team help by proofreading a draft version and identifying problems, but the words and content contained in the document should be your own.

### Submission Guidelines

Turn in **a** printed copies of your Requirements Specification Document. Include a link to your Requirements Specification Document on your team's Web site. I will download the file from your site to help with grading.

Ask, and if desired, provide your faculty advisor and/or customer with a printed copy of your Requirements Specification Document.