

CSC 493 Senior Projects

R2: Weekly Report 2 with Scope Document & Preliminary Requirements Specification

On the structure, format, and purpose of the weekly reports

Be sure you have read http://faculty.berea.edu/pearcej/csc493/R_OnReports.pdf because you will be expected to follow the guidelines stated there for each weekly report. In particular, each weekly report must be formatted as described there and must include the cover page, the Application Development section, and the Executive section as described there.

Summary of the new tasks for inclusion in the Application Development Section

This week new work will involve creating two new sections which are related to one another, a Vision and Scope Document, as well as a Preliminary Software Requirements Specification (SRS).

We have a very constrained time-frame in this course to complete a sizeable project. This means that scaling the scope of your project is very important. In English, the word “scope” means the range, size, or extent of something. In software development, one creates a Vision and Scope Document or a Scope Document to clearly define the scope of the project, i.e. to clearly specify what the software developers understand to be their tasks. This means that “out of scope” tasks are what the software developers assume are NOT their responsibilities. Note that even though it is called a Scope Document, it is typically just a small section of a larger design document, as it will be in our case. In the business world, clarifying the scope and limitations helps to establish realistic expectations of the stakeholders and provides a reference frame against which proposed features and requirements changes can be evaluated.

It has been long recognized that the hardest part of building a software system is deciding precisely what to build. Thus, no other part of the work is as difficult as establishing the detailed technical requirements. The iterative development model of software engineering employs the construction of very small components which are tested and then built upon with additional small components. Iterative process models are often used by commercial developers because they are models which allow flexibility in achieving design goals of a customer who does not know how to explicitly describe what they need. The specific iterative software engineering model recommended in this course is a Joint Application Development (JAD) model which is a methodology of evolutionary prototyping that assumes project requirements are not completely fixed when the project begins but may emerge more fully and even change as the project develops -- up until the time they must be finalized in order to complete the project.

A Software Requirements Specification (SRS) is a written understanding of system requirements prior to any actual design or implementation work. The SRS document states in precise and explicit language those functions and capabilities a software system must provide, and any required constraints which the system must follow. An SRS contains functional and nonfunctional requirements only, rather than design ideas. Functional requirements describe the functionality or capability that the software is to have. Example functional requirement: "Software must update and remember the highest score earned by all users." Non-functional requirements are qualities or constraints that the software must have or comply with, but which are not operations that will be automated. Example non-functional requirement: "Software must run on Windows XP". Requirements are typically prioritized on as well as numbered so that they can be tracked as the project moves forward.

Application Development Section of the Report:

The Application Section of the report should be divided into the following sections.

- **The Project Concept Proposal**
- **Inspiration**
- **Vision and Scope**

Please limit this new section to two paragraphs.

- The first paragraph in this section should present a concise vision statement which summarizes the purpose and intent of the project and describes what the world will be like when the project is completed. This vision statement may be somewhat idealistic, but it still should be grounded in the realities of time and resource limitations.
- The second paragraph in this section should very clearly specify both the scope of the project as what will be considered “out of scope”. Be sure to describe the intended major features that will be included in the project. Also, identify any product features or characteristics that a stakeholder might anticipate, but which are not planned to be included in the project.

- **Preliminary Software Requirements Specifications**

Using your Scope Document as a guide, precisely and explicitly describe all of the primary software requirements using the following template for each specific requirement:

- Number: (List unique requirement number.)
- Statement: (Precisely state the requirement.)
- Evaluation Method: (How can you tell if the completed software satisfies this requirement?)
- Dependency: (List each other requirement on which satisfaction of this requirement depends or write "None")
- Priority: (Assign a priority to this requirement: essential, high, middle, low, or if time permits.)
- Requirement revision history: (when, what, and why)

Be sure that you list each requirement separately rather than combining two or more.

Report Format and Submission:

Convert the file to a pdf, name the file *yourusername-R2.pdf*, and submit into our course management system in time to meet the stated deadline.

Note that the pdf file does not need to be printed, but you will need to retain the electronic copy of the original editable file because you will use it to build your next weekly report.