

# **TITLE**

## **Senior Project Requirements**

*Fall 2018 & Spring 2019*

*Peter C. Chapin <[pchapin@vtc.edu](mailto:pchapin@vtc.edu)>*

### **Introduction**

*Give a brief overview of your project. Explain what your project will do and what problem it solves. This section should be very short, maybe just one paragraph. You've already covered this ground with your proposal so you can probably just take some text from that earlier document.*

### **Functional Requirements**

*Describe what your project is going to do in concrete, measurable terms. Categorize your requirements to distinguish the essential ones from the less essential ones. Use cases are often a good way to capture functional requirements, but other methods can be effective as well. Consider the actors who will be interacting with your project, and be sure to comment on your system's user interface. This is also a good place to identify milestones.*

*It is usually a good idea to give individual requirements names so you can talk about them later and so you don't have to worry about renumbering them if (when) you make changes to your list of requirements.*

### **Non-Functional Requirements**

*Describe requirements that are not related to the actual operation of your project. Use the headers below to guide you but don't feel slavishly bound to them.*

#### **Platform**

*What types of computer systems does your project need to support? Which operating systems will you target? Are you going to build a mobile version of your system?*

#### **Performance**

*What is the minimum amount of resources that your project will require? Give specific numbers if possible.*

#### **Security**

*What kinds of attacks must your product be able to resist? What kinds of attacks will your product not necessarily protect against?*

#### **User Characteristics**

*Are the users technically knowledgeable? What skills do you assume the users have?*

## **Scale**

*How large a system must your project support? (number of users, number of files, number of simultaneous network connections, as appropriate)*

## **Documentation**

1. *Design Documentation.* This document will be a detailed design that describe how the various components of the system work. It will include block diagrams, schematics, or code snippets, as appropriate. The audience of this document is other developers who might want to understand and extend the system.
2. *User Documentation.* This document describes how to install, configure, and use the system. The audience of this document is people who want to use your system.
3. *Web Site.* A web site will be produced that describes the system. The audience of this site is potential users considering the system and current users looking for technical support. This site can be hosted anywhere that is convenient. You can use the departmental Linux host lemuria.cis.vtc.edu (PHP and various database servers available), or you can use some other web host of your choice.

## **Data Formats**

*What kind of input/output formats must your project be able to handle? This is particularly relevant if there are appropriate standard formats to consider.*

## **Internationalization**

*Does your project need to support multiple languages or cultures? If it does not, it is best to say so explicitly.*

## **Environment**

*In what sort of physical environment will your project operate? Is it required to work in adverse conditions such as high or low temperatures, under water, or high vibration?*

## **Expected Enhancements**

*What sort of enhancements are likely? The answer to this question may impact how your system is organized.*

## **Date**

The final presentation will be on the last day of classes of the Spring 2019 semester. The final demonstration (if needed) and the documentation will be due at the end of finals week.