[Product Name]

[Project Team Members]

Software Requirements Specification

Draft X
Date

CSE 403 - DEMS

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Number	Full Name	Information about the revision.	00/00/00

Overall Description

Description

Identify the software product to be produced. Explain what the product will and will not do. Describe the relevant benefits, objectives, and goals of the product. Identify the target customer.   We'd expect your proposal document would have material relevant for this section, although you will have significantly shaped the project since that point.

Identify the documentation that will accompany the software product.

Scope

Describe the limits of the system, including what it will *not* do.

Describe the hardware and software that the product depends on/will operate on. If there are performance or reliability parameters, include those.

Use Cases (OPTIONAL)

Provide at least:

1. One (1) use case summary diagram (stick diagram) that depicts several of the major use cases and actors that are important in your system, including the use cases described in 2.

2. Two (2) formal use cases for scenarios you think are two of the most important to your product. They should be similar to Use Cases from Cockburn's book excerpt and the class exercise [note: I may not do this], and should include: primary actor, level, preconditions, success and failure end conditions, trigger, a list of steps to the success scenario, a list of properly numbered extensions, and a failure-handling remedy for each extension as appropriate, and a list of variations. It is impossible to think of every possible failure case and how to solve it ahead of time. Therefore it is understood that you may miss a few extension cases, and that you may not have a completely detailed remedy for each extension. However, your list of extensions should reflect extensive thought about the subject, and you should have failure remedies if reasonable ones exist. If you do not have a known remedy for an extension, your use case should state this clearly and explain why this is the case and what will be done to investigate possible remedies.

Feature List

Describe the features that you will deliver at the various release points of your product (beta, final). (No *features* are required for the Zero Feature Release; just infrastructure.) Think of these as commitments to DEMS. If there are features that you will strive for but that may not make it into the final release, mark them as “stretch”. Ultimately, you and your customer will need to agree on the feature list and the proposed delivery times, and later negotiate if there is to be movement.

This is the start of a feature list for a networked calendar project. Your feature list should have about 15 of the features you expect to deliver.

Feature	**Target**
Create/delete a new calendar	Beta
Create/delete a new event	Beta
View a friends calendar	Final
Add events to a friends calendar	Final
Create/delete repeating events	Final
Reminder email and popup (configurable) when event occurs	Final
Import Outlook events into calendar	Stretch

UI Prototype

Provide diagrams containing rough sketches of your product's user interface. The diagrams can be hand-drawn, drawn by computer, or can come from screenshots of an actual programmed prototype if you like. If a window leads to a dialog box, drop-down box, etc., perhaps this should be included as a sub-diagram. Your diagrams do not need to be pretty to get full credit, but they should be legible and reflect some forethought about what options will need to be shown and how the user will use the software.