On-line Co-op Evaluation System

Software Requirements Specification

Char c = 5
Patrick Flanagan
Tom Small
Whitney Sorenson
Dan Volpe
Chris Woodbury

May 10, 2005

Table of Contents

1. Introduction....2
1.1 Purpose....2
1.2 Intended Audience....3
2. Product Description....3
2.1 Purpose....3
2.2 Users....3
Students....3
OCECS Representatives....3
Academic Department Representatives....3
2.3 Assumptions....3
3. Product Context....4
3.1 Diagram....4
3.2 Workflow....5
4. External Interface / Look and Feel Requirements....5
4.1 External Interface Requirements....5
4.2 Look and Feel Requirements....6
5. Functional Requirements....7
6. Non-functional Requirements....12
6.1 Performance....12
6.2 Safety....12
6.3 Security....13
6.4 Legal....13
6.5 Cultural and Political....13
6.6 Quality Attributes....13
6.7 Business Rules/Relevant Facts....14
6.8 User Documentation....14
6.9 Design and Implementation....15
8. New Problems in Existing Customer Environment....17
9. Cutover....17
10. Risks....17
11. Costs....19
12. Future Releases....19
13. Appendices....19
13.1 References....19
13.3Glossary....29

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) formally describes the On-line Co-op Evaluation System, herein referred to as “the system.” It provides a detailed description of the system to the customer, decomposition of the problem, a basis for the design of the system, and a basis for testing the system once it’s completed. It establishes the external interface, functional, and nonfunctional requirements for the system. This document only

covers the extensions to the system that are being made to allow student co-op evaluation submission online.

1.2 Intended Audience

This document is intended as a means to understand the system for the:

- Customer
- Prospective users
- Developers
- Faculty advisors
- Other stakeholders

2. Product Description

2.1 Purpose

The Online Co-op Evaluation System was created last year by another Senior Project team. That team only implemented the functionality for employers to submit feedback online, the new system needs to allow students to do that as well. This will eliminate the need for students to mail in paper versions of their co-op evaluations. Academic users will then be able to search and view these submissions online and administrative users can create new forms to be sent to students. They will also be able to generate reports based on these submissions, giving them insight into what students think.

2.2 Users

Students

Students will use the Online Co-op Evaluation System in order to submit evaluations of their co-op experience. They will generally use the system once per quarter while they are on co-op, although they may make changes to their forms at a later time.

OCECS Representatives

OCECS representatives will still use the Online Co-op Evaluation System to do everything the old system did. In addition, they will be able to manage new forms for students to fill out on co-op and generate reports based on data submitted by the students. They will also handle changes to student forms that are requested by departments.

Academic Department Representatives

Academic department representatives will use the Online Co-op Evaluation System to process student evaluation forms. They will use individual form data in order to grade students for their co-op experience. They will also be able to search for a student's past co-op reports and see a history of what has been done.

2.3 Assumptions

The team is making the following assumptions for the project:

- OCECS Administrators have DCE login accounts that will be used for authentication for this system.
- Department users have DCE login accounts that can be used to login and identify their department.
- The hardware shall be supplied by the Co-op Office or use ITS' current hardware.
- The web server can process the input of form data.
- The email server can handle sending over 1000 messages at a time.
- The ITS department will maintain the system after the project has completed.

3. Product Context

3.1 Diagram

The diagram below shows the components of the system and those with which it interacts. They include the ITS DNS (domain name server)/web server, the DCE LDAP server for user authentication, the ITS Oracle database instance for the system's data, and the ITS mail server for sending electronic mail.

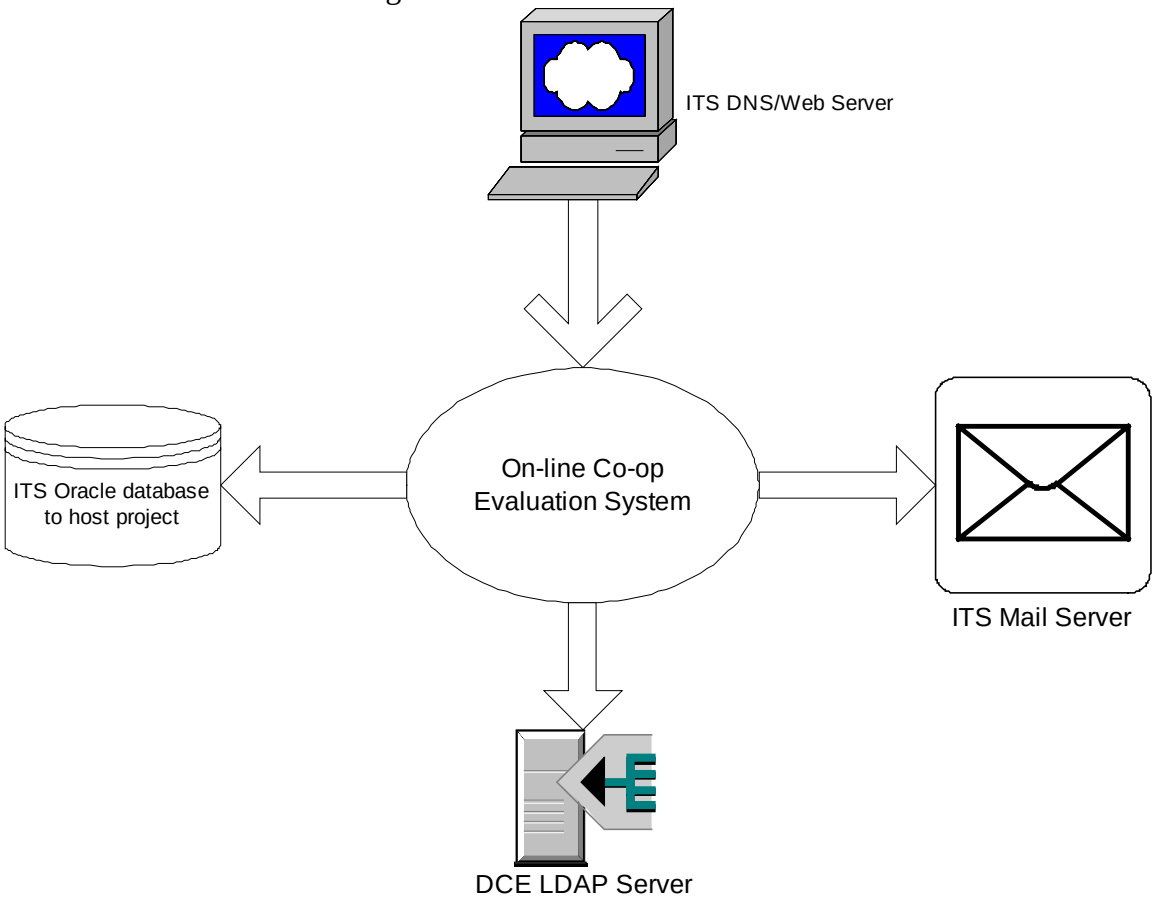

Figure 1 - Context Diagram

3.2 Workflow

The workflow for the additions to the system is shown with the diagram in Figure 2. This diagram shows the tasks that OCECS representatives, students, and department users will use the system for. The users will still be able to perform all the tasks from the original system, but only the new tasks are shown below.

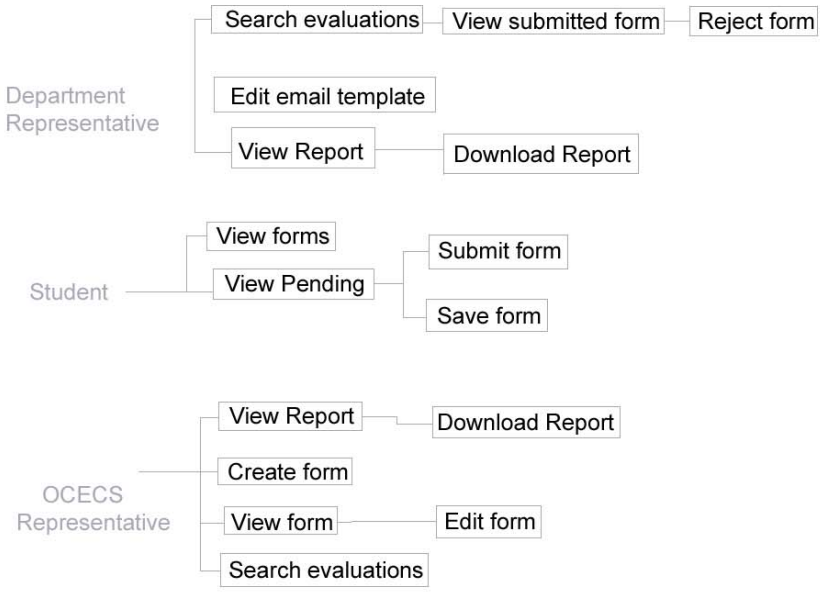

Figure 2 - Workflow Diagram

4. External Interface / Look and Feel Requirements

4.1 External Interface Requirements

The extension should work with all the external interfaces as the existing employer evaluation system in addition to these.

Requirement ID	EI-1
Requirement Type	Software Interfaces
Description	Interface with DCE for user validation purposes via LDAP
Source	MN 1-6-2005
Fit Criterion	The system can authenticate user logon credentials via DCE for students.
Priority	High

Requirement ID	EI-2

Requirement Type	Communications Interfaces
Description	Interface with ITS mail servers.
Source	Statement of Need
Fit Criterion	The system can send mass email via the ITS mail server. This will involve sending both messages to single users and messages to groups of users.
Priority	High

Requirement ID	EI-3
Requirement Type	Data format
Description	The data generated by reports needs to be exportable into a format that that OCECS and academic department users could import into a spread sheet.
Source	MN 12-7-2004
Fit Criterion	Many users like to work with the data in a spreadsheet to draw useful data from it. This means that the data should be exportable into a comma-separated text file or other Excel-friendly format.
Priority	High

Requirement ID	EI-4
Requirement Type	Communications Interfaces
Description	Method for getting student info from co-op database
Source	MN 2-8-2005
Fit Criterion	The system should have an automated way of getting a student's information from the OCECS database into the student co-op evaluation database.
Priority	High

4.2 Look and Feel Requirements

The extension should satisfy all the previous look and feel requirements as well as these.

Requirement ID	LF-1
Requirement Type	Look and Feel Requirements
Description	The interface should not rely solely on colors to highlight things that need attention.
Source	MN 2-8-2005
Fit Criterion	Some users of the system may be colorblind, so there should be a way to highlight items that need attention in some ways other than simply changing their color.
Priority	High
Requirement ID	LF-2
Requirement Type	Look and Feel Requirements

Description	Tables should be consistent with respect to what columns are displayed
Source	MN 3-14-2005
Fit Criterion	Search results for evaluations should be ordered alphabetically by student last name. Returned data should be sortable by a column by clicking on the column header in their browser. The system should also show whether the evaluation is saved, submitted, or pending.
Priority	High

5. Functional Requirements

The extension of this system should contain all the existing functionality in the employer evaluation system in addition to these.

Requirement ID	FU-01
Requirement Type	Functional
Use Case ID	UC-1
Description	Students need to be able to login with their DCE account
Source	Statement of Needs
Fit Criterion	Students should be able to authenticate with the system using their DCE accounts.
Priority	High

Requirement ID	FU-02
Requirement Type	Functional
Use Case ID	UC-9
Description	The system needs a way to alert students that they have something to do.
Source	MN 12-7-2004
Fit Criterion	The system should send students an email to alert them that they need to take some action in the system. These emails can tell the student about things such as upcoming deadlines, incomplete forms, or new forms to fill out.
Priority	High

Requirement ID	FU-03
Requirement Type	Functional
Use Case ID	UC-9
Description	Emails should be staggered to reduce load on the email server.
Source	MN 1-6-2005
Fit Criterion	The emails that the system sends out for alerts and notifications should be done at an off-peak time and staggered so as not to overwhelm the mail server.
Priority	High

Requirement ID	FU-04
Requirement Type	Functional
Use Case ID	UC-5

Description	Email notifications should be viewable by departments
Source	MN 2-1-2005
Fit Criterion	Departments should have the ability to view the contents of a section of the email to students and to generate a test email.
Priority	Medium

Requirement ID	FU-05
Requirement Type	Functional
Use Case ID	UC-2
Description	System should send confirmation message to students
Source	MN 12-14-2004
Fit Criterion	After a student has successfully submitted their co-op evaluation, the system should email that student a confirmation message and display a confirmation message on screen.
Priority	High

Requirement ID	FU-06
Requirement Type	Functional
Use Case ID	UC-9
Description	System should log email dates
Source	MN 2-8-2005
Fit Criterion	The system should keep a log of what emails were sent to students so that administrators can see when the last action on an evaluation was.
Priority	High

Requirement ID	FU-07
Requirement Type	Functional
Use Case ID	UC-7, UC-8
Description	Forms will need new question types
Source	MN 2-1-2005
Fit Criterion	Forms will need to support the same types of questions as the existing employer forms. There is also a new question type, the double likart, which allows comments. There are also essay questions where the text is captured.
Priority	High

Requirement ID	FU-08
Requirement Type	Functional
Use Case ID	UC-2, UC-3, UC-7, UC-8
Description	Each department can have its own form for students to fill out.
Source	MN 12-2-2004, MN 2-1-2005
Fit Criterion	Each department can have a custom form for students to fill out. These forms will contain questions from the co-op office and questions from the department. The Co-op Office administrations will be able to edit the forms for each department.

Priority	High

Requirement ID	FU-9
Requirement Type	Functional
Use Case ID	UC-2
Description	Students should be able to save partially completed forms
Source	MN 12-14-2004
Fit Criterion	When an incomplete form is submitted, it will be saved so that the student can return and complete it at a later time. When the student returns to a saved form, the questions remaining should be highlighted using both text color and a marking. The questions could be un-highlighted as they are filled out.
Priority	Medium

Requirement ID	FU-10
Requirement Type	Functional
Use Case ID	UC-3
Description	Students should be able to see past evaluations
Source	MN 12-14-2004
Fit Criterion	Students should be able to view past co-op evaluations that they have submitted as well as evaluations that their employers have submitted about them. They should be in the same place, so that they can both be viewed at the same time.
Priority	High

Requirement ID	FU-11
Requirement Type	Functional
Use Case ID	UC-3
Description	Students should be able to check if employers have submitted evaluations yet
Source	MN 12-14-2004
Fit Criterion	Students should be able to use the system to see if their current employer has submitted an evaluation for their current co-op yet.
Priority	High

Requirement ID	FU-12
Requirement Type	Functional
Use Case ID	N/A
Description	Employer forms should not be stored twice
Source	MN 2-1-2005
Fit Criterion	In the existing employer evaluation system, both the saved version and the submitted version of an employer's evaluation are stored in the database, which is causing the database size to grow unnecessairly. It would be nice if this could be fixed.
Priority	Low

Requirement ID	FU-13

Requirement Type	Functional
Use Case ID	UC-5
Description	Administrators should be able to view and modify department email templates.
Source	MN 2-8-2005
Fit Criterion	Co-op administrators should be able to view the email templates that departments set up to use as well as change the text.
Priority	Medium

Requirement ID	FU-14
Requirement Type	Functional
Use Case ID	UC-4, UC-6
Description	Search should support partial names
Source	MN 3-7-2004
Fit Criterion	When performing a search, users should be able to enter a partial name and get a list of all students that match it.
Priority	High

Requirement ID	FU-15
Requirement Type	Functional
Use Case ID	UC-4, UC-6
Description	Users should be able to search employer evaluations by company
Source	MN 3-7-2005
Fit Criterion	Users doing a search should be able to search for employer evaluations by company name.
Priority	High

Requirement ID	FU-16
Requirement Type	Functional
Use Case ID	UC-4, UC-6
Description	Users should be able to search for students by student ID number
Source	MN 3-7-2005
Fit Criterion	Some departments don't use DCE for anything and only reference students by their student ID numbers. The system needs to support searching for students by both student ID and DCE account. There are privacy concerns since the student ID number is usually that student's social security number.
Priority	High

Requirement ID	FU-17
Requirement Type	Functional
Use Case ID	UC-4
Description	Department users should be able to reject forms
Source	MN 3-7-2005

Fit Criterion	After a form has been submitted, a department user can review it and optionally reject it. If a form is rejected, it's status should be set back to saved and an email should be sent to that student telling them that their form has been rejected.
Priority	High

Requirement ID	FU-18
Requirement Type	Functional
Use Case ID	UC-6
Description	The system shall allow OCECS representatives to create reports containing data from submitted evaluations.
Source	Statement of Need
Fit Criterion	A report may be generated that contains data from submitted evaluations.
Priority	High

Requirement ID	FU-19
Requirement Type	Functional
Use Case ID	UC-6
Description	System should be able to generate reports based on student data
Source	MN 3-7-2005
Fit Criterion	Reports should be able to group likart scale data over a specified range, view student work reports by specified grouping (e.g. all in a specific quarter), select groupings of quarters instead of just one quarter, export the data into a spreadsheet, and get a summary of information for one student.
Priority	High

Requirement ID	FU-20
Requirement Type	Functional
Use Case ID	UC-4, UC-6
Description	System should support searching on multiple criteria
Source	MN 3-14-2005
Fit Criterion	The search page should allow users to search on multiple selected majors, search by company, and be able to search based on submitted evaluations, saved evaluations, pending evaluations, or all evaluations.
Priority	High

Requirement ID	FU-21
Requirement Type	Functional
Use Case ID	UC-10
Description	The system should support storing waived co-ops
Source	MN 3-14-2005

Fit Criterion	The system should support departments waiving a co-op. It should store a default page for waived forms and display the name of who authorized the waiver as the co-op contact name.
Priority	Low, not implemented

Requirement ID	FU-22
Requirement Type	Functional
Use Case ID	UC-1
Description	Student page should show remaining co-ops
Source	MN 3-14-2005
Fit Criterion	The student view page should display how many co-ops a student has left to do. If the student is a senior and has co-ops left, it should display an alert telling them that they still have co-ops to complete.
Priority	Low

6. Non-functional Requirements

6.1 Performance

The extension to the system should satisfy the same performance requirements as the original system and the new additions should be hinder the system's performance.

6.2 Safety

The extension to the system should satisfy all the original safety requirements in addition to these.

Requirement ID	SA-1
Requirement Type	Safety
Description	The system shall prevent users from accidentally losing the data they have entered before it's submitted.
Source	Assumption
Fit Criterion	The system shall provide confirmation if a user cancels an operation before submitting data or navigating to a different page.
Priority	High

6.3 Security

The extension should meet all the security requirements of the existing employer evaluation system in addition to these.

Requirement ID	SE-1
Requirement Type	Security
Description	The system should not allow students to view other student's evaluations
Source	Assumption

Fit Criterion	Students should only be able to view their own evaluations and not those of other students.
Priority	High

6.4 Legal

The extension should meet the same legal requirements as the existing employer evaluation system. There are no new legal requirements for the system extension.

6.5 Cultural and Political

The extension should meet the same cultural and political requirements as the existing employer evaluation system. There are no new cultural or political requirements for the system extension.

6.6 Quality Attributes

The extension should meet the same quality requirements as the existing employer evaluation system in addition to these.

Requirement ID	AV-1
Requirement Type	Availability
Description	The system shall be online for 23 hours a day, 7 days a week
Source	Assumption
Fit Criterion	During testing the system stays online for 23 hours a day, 7 days a week. The system is allowed to do down for 1 hour a day for maintaince.
Priority	High

Requirement ID	AV-2
Requirement Type	Availability
Description	The system shall be available from any computer with an internet connection
Source	Assumption
Fit Criterion	
Priority	High

Requirement ID	MA-1
Requirement Type	Maintainability
Description	The system shall need no more than 1 hour a week of maintenance, not including importing of data
Source	Assumption
Fit Criterion	At no point will more than an hours worth of maintance need to be done to the system. The new changes to the system should not require more maintaince time than this.
Priority	High

Requirement ID	EX-1
Requirement Type	Expandability
Description	The system shall allow new evaluation forms to be added

Source	Assumption
Fit Criterion	The system should allow co-op administrators to add new forms to the system for a specific department.
Priority	Medium

Requirement ID	SY-1
Requirement Type	Scalability
Description	The system shall allow new colleges to be added to the system
Source	Assumption
Fit Criterion	New colleges can be added to the system
Priority	High

Requirement ID	SY-2
Requirement Type	Scalability
Description	The system shall allow new departments to be added to the system
Source	Assumption
Fit Criterion	New departments can be added to the system
Priority	High

6.7 Business Rules/Relevant Facts

The extension will need to take into account the same relevant facts and business rules as the existing employer evaluation system in addition to these.

Requirement ID	BR-1
Requirement Type	Relevant Facts
Description	A student on co-op must have their co-op registered with the Co-op Office to be able to fill out evaluations online.
Source	Assumption
Fit Criterion	If a student is registered, there should be a co-op evaluation form in the database waiting for them to fill out.
Priority	High

6.8 User Documentation

Requirement ID	UD-1
Requirement Type	User Documentation
Description	The user manual must be consistent in all wordings and grammar.
Source	Assumption
Fit Criterion	All references to the same concept or item will use the same terminology. No mispelled words shall be found.
Priority	Med

Requirement ID	UD-2
Requirement Type	User Documentation

Description	The user manual will contain a getting started section to describe the setup process.
Source	Previous user manual
Fit Criterion	Users can setup the system without problems by using the provided manual.
Priority	Med

Requirement ID	UD-3
Requirement Type	User Documentation
Description	The user manual will be broken into sections to separate operations and tasks that are implemented by the system.
Source	Assumption
Fit Criterion	The manual is organized and users can easily perform any task, which will be evaluated by a survey.
Priority	Med
Requirement ID	UD-4
Requirement Type	User Documentation
Description	The user manual will need to be updated to describe the new functionality added to the system.
Source	Assumption
Fit Criterion	The manual is organized and users can easily perform any task, which will be evaluated by a survey.
Priority	Med

6.9 Design and Implementation

Requirement ID	DI-1
Requirement Type	Design Constraint
Description	ITS requires use of Java
Source	MN 12-2-2004
Fit Criterion	ITS requires the use of Java for this project. They use JBoss as the Java Application Server, and want this project to use it as well.
Priority	High
Requirement ID	DI-2
Requirement Type	Design Constraint
Description	User Authentication
Source	MN 12-2-2004
Fit Criterion	The system will use DCE accounts for authentication when logging in.
Priority	High
Requirement ID	DI-3
Requirement Type	Implementation Constraint

Description	Database backend
Source	MN 12-2-2004
Fit Criterion	ITS requires the use of Oracle for the database. They will provide the team with an Oracle database.
Priority	High
Requirement ID	DI-4
Requirement Type	Implementation Constraint
Description	Build System
Source	MN 12-2-2004
Fit Criterion	ITS requires ANT scripts to build and deploy the system.
Priority	High
Requirement ID	DI-5
Requirement Type	Implementation Constraint
Description	Use of client-side functionality
Source	MN 12-2-2004
Fit Criterion	The system should not rely on JavaScript or cookies for important functionality since they are both a security and reliability concern. All state should be stored on the server side.
Priority	High
Requirement ID	DI-6
Requirement Type	Implementation Constraint
Description	Hosting
Source	MN 12-2-2004
Fit Criterion	ITS will maintain the server that is running the web server and database.
Priority	High

8. New Problems in Existing Customer Environment

With any new system, there is always the potential for creating new problems in an existing environment.

1. The On-line Co-op Evaluation System will impact the Co-op Office because they will need to analyze the data submitted by students. They will also need to work more closely with departments in order to customize co-op evaluation forms. They have already made process changes in order to accommodate using the existing system, so the change will not be as radical for them. The OCECS representatives will need to become familiar with the changes made to the system, but since they are already familiar with the current system it will be easy to adapt.
2. With changes made to the system, it will get a lot more traffic with students visiting the site. Additionally, the database will grow in size each quarter as it stores all the student evaluations. ITS will be in charge of maintaining this

system and they will have to determine if the current hardware is not meeting expectations.

3. Forms are allowed to be edited, and academic departments can request that OCECS representatives make changes to their existing forms. However, once a form has been delivered to students to fill out, the form for that quarter cannot be changed.
4. Academic departments will be affected by this new system, since it will replace the current paper submissions. Academic users will need to be trained in the use of the new system and begin using the new system for evaluating student work reports instead of doing it on paper.

9. Cutover

There is already data in the employer database, and we need to integrate the student database with that. The team has been given a copy of the employer database and has been making structural changes to it. These changes will be given to ITS to incorporate into the production database.

10. Risks

The On-line Co-op Evaluation System has several risks associated with the project. Earlier risks were listed in the old SRS document, and risks that are specific to the extension of the system are listed here.

1. **Database data loss:** With databases, there is always the possibility of loosing data, either by corrupted data or by accidental deletion. By creating a backup database, in which copy of the main database will be main on a schedule of some kind, the hopes of always having a current backup database is in the planning. ITS has a backup program they use for databases, which will hopefully prevent any data loss.
2. **Search Capabilities:** Finally the biggest risk in the deployment of the On-line Co-op Evaluation System is that it might now support all the search capabilities the Co-op department may want. The system will contain all possible ways of doing a form search or analysis based on features mentioned by members of the Co-op office before the deployment of the On-line Co-op Evaluation System.
3. **Introducing bugs into existing employer system:** Since there is already an existing system in place, there is the possibility of introducing new bugs into the system. To minimize the risk of introducing new bugs, not only should new functionality be testing but all the existing functionality as well.
4. **Configuration of existing employer system:** Since none of the current team is familiar with the existing system, the team needs to work closely with ITS to determine how the system is currently configured to get everything set up.
5. **Changing Requirements:** The stakeholders may not have a clear idea of how exactly the system should work or what it will look like. The system should be designed with this in mind, since requirements will likely change during development.
6. **Difficulty understanding existing employer system design:** The existing system is complex, and the current team is not familiar with it. The original team

left documentation describing how the system works, so time should be allocated to go through this documentation and get an understanding of how things currently work.

7. **Integration Problems with existing employer system:** The extensions to the current system are not stand-alone; they will need to be integrated into the existing system. Time needs to be allocated for testing the integration of all the components.
8. **Keeping System Secure:** Since some team members are not familiar with Oracle or JBoss, the team should work closely with ITS to ensure a secure environment is created and that it is compatible with the existing security in place.
9. **Testing/Debugging:** The system needs to work right, and it needs to do what the users want it to do. Testing should be done on both the functionality level and the usability level.
10. **Tight Schedule:** The schedule for the project is aggressive, and it must be completed on time. There are 6 months allocated for the project, and the project must be delivered by then.
11. **ITS contact problems:** Since ITS is involved in a lot of aspects of the project, they need to be involved during development for questions and technical issues. If they are unavailable or too busy to respond to requests, that can hurt the project.
12. **Design complexity:** The design is fairly complicated, so it needs to be documented for both developers and the future maintainers of the system.
13. **Installing tools locally:** Some of the tools that are needed for development may not be free, so the team should spent time looking for tools to use up front and arrange with ITS to have any required software. The team also needs to spend some time using the tools to ensure that they understand them.

11. Costs

There are no costs for us in this project. It is a school-sponsored project; therefore, there is no cost for man-hours of work involved in the development of this project. Hardware for the servers are being provided to us by the Co-op office as well is ITS. All software used is free from companies that do not have license fees or registration fees.

12. Future Releases

Functionality for future releases will go here. At this point, it's not clear what will be needed for future releases.

13. Appendices

13.1 References

1. Original Online Co-op Evaluation System SRS
2. Statement of Needs
3. Notes from meetings with stakeholders

13.2 Use Cases

The use cases here are only for modified functionality. If a use case is not shown here, it is unchanged from the original system.

Use Case ID: UC-1
Use Case Name: StudentLogsOn
Primary Actor: Student
Goal: To log into the system to take other actions

Pre-Conditions:
- The student has a valid DCE account

Post-Conditions:
- The user is logged on to the system
- The user is at the main page

Main Scenario:

Step	Actor Action	Step	System Reaction
1	The user browses to the log on page		
2	The user enters the username/password		
3	The user submits the information	4	System verifies that the username/password is correct
		5a	System loads the main page
		5b	System informs the user the username/password is incorrect

Exceptions: none
Use Cases Utilized: none
Notes and Issues: none

Use Case ID: UC-2
Use Case Name: StudentViewsPendingForms
Primary Actor: Student
Goal: To view and fill out a Co-op Work Report

Pre-Conditions:
- The user is logged on to the system
- The user has an evaluation to fill out

Post-Conditions:
- The form information is entered into the database
- The main page is displayed

Main Scenario:

Step	Actor Action	Step	System Reaction
1	The user selects a pending work report	2	The system displays the work report, including any answers that the student has already filled out.
3	The user fills out the work report		
4a	The user submits the information	5a	The system stores the information in the database and marks it as submitted. The system then sends a confirmation email
4b	The user saves the information	5b	The system stores the information in the database but does not mark it as submitted
		6	If the user has selected to change co-op information, an email is generated and sent to the co-op office
		7	The main page is displayed

Exceptions:
- If the user does not have a form to fill out
  - The system does not display a link to pending forms
- If the information submitted is invalid
  - The system displays an error page with the invalid information and returns to the form

Use Cases Utilized: none
Notes and Issues: none

Use Case ID: UC-3
Use Case Name: StudentViewsSubmittedForms
Primary Actor: Student
Goal: To view previously submitted evaluations and work reports

Pre-Conditions:
- The student is logged in
- There are submitted evaluations for the student

Post-Conditions:
- The student has seen the evaluations and work reports

Main Scenario:

Step	Actor Action	Step	System Reaction
1	The user selects to view submitted evaluations	2	The system displays a list of all the evaluations and work reports for this student in the database
3	The user clicks on a specific evaluation or work report	4	The system will display the evaluation or work report with the values filled in.

Exceptions:
- If the student has no saved evaluations or work reports
- The system will not display a link to any reports or evaluations

Use Cases Utilized: none
Notes and Issues: none

Use Case ID: UC-4
Use Case Name: ViewEvaluations
Primary Actor: Department Representative, OCECS Representative
Goal: To obtain a list of submitted evaluations and view them individually if desired

Pre-Conditions:
- The department user is logged in
- The main page is displayed

Post-Conditions:
- The department user has been the evaluations

Main Scenario:

Step	Actor Action	Step	System Reaction
1	The user selects to search evaluations	2	The system allows the user to input search criteria
3	The user enters search criteria for what they want to see	4	The system displays a list of all the matching evaluations with a link to see them
5	User can optionally click a link to see the submitted evaluation	5	If a specific link is clicked, the system will display that evaluation to the user.
5a	User selects to reject the evaluation	6a	The system will mark the evaluation as saved as send an email to the student altering them that it has been rejected.
5b	User selects to return to search page	6b	The search page is displayed again allowing the user to do another search

Exceptions:
- The user is not associated with a department in the database

Use Cases Utilized: none
Notes and Issues: none

Use Case ID: UC-5
Use Case Name: EditDepartmentEmailTemplate
Primary Actor: OCECS Representative
Goal: To change the layout of the email template departments use

Pre-Conditions:
- The user is logged into the system
- The user has the appropriate privileges to edit the template

Post-Conditions:
- The changed template is stored in the database

Main Scenario:

Step	Actor Action	Step	System Reaction
1	The user selects to edit the template	2	The system displays a list of the templates that the user is allowed to edit
3	The user makes changes to the template		
4a	The user saves the changes	5a	The changes are saved into the database
4b	The user cancels the edit	5b	The system does not save the changes
		6	The system displays the main page

Exceptions: none
Use Cases Utilized: none
Notes and Issues: none

Use Case ID: UC-6
Use Case Name: ViewStudentDataReport
Primary Actor: OCECS Representative, Department Representative
Goal: To generate a report based on feedback from student evaluations

Pre-Conditions:
- The OCECS or academic representative is logged on to the system
- There are student work reports submitted to generate data from

Post-Conditions:
- The user has seen the report

Main Scenario:

Step	Actor Action	Step	System Reaction

1	The user selects to generate a report	2	The system allows the user to enter what criteria they want for the report
3	The user enters the report criteria	4	The system generates a report based on what they user specified
		5	The system displays the report results to the user
6a	The user can selects to download the results in a spreadsheet	7a	The system sends the user the data in a spreadsheet format
6b	The user selects to return to the main page	7b	The system does not send the user any extra data
		8	The main page is displayed

Exceptions:

- There are no submitted work reports
- An error message should be displayed alerting the user that there is no data

Use Cases Utilized: none
Notes and Issues: none

Use Case ID: UC-7
Use Case Name: CreateNewForm
Primary Actor: OCECS Representative
Goal: To create a new form for departments

Pre-Conditions:
- The OCECS representative is logged on to the system

Post-Conditions:
- None

Main Scenario:

Step	Actor Action	Step	System Reaction
1	The user selects the create forms page	2	The system displays a list of the forms that this user has access to
3a	The user selects a department and clicks Create Form	4a	The system creates a blank form for the specified department and quarter
3b	The user selects a department and specifies an XML file to import	4b	The system creates a form for the specified department and quarter already populated with the questions in the XML file.
		5	The system displays the department forms page

Exceptions:
- Formatting error in the XML file
  - If there is an error in the XML file, the system should display a message and not create a form

Use Cases Utilized: none
Notes and Issues: none

Use Case ID: UC-8
Use Case Name: EditForm
Primary Actor: OCECS Representative
Goal: To add, change, or delete questions from an existing form

Pre-Conditions:
- The OCECS representative is logged on to the system.
- The OCECS representative has permission to edit forms

Post-Conditions:
- The changed form is saved into the database

Main Scenario:

Step	Actor Action	Step	System Reaction
1	User selects the forms page	2	The system displays a list of all the forms that the user has permission to edit.
3	The user selects a form to edit	4	The system displays the form with links for adding questions, editing questions, or deleting questions
5a	User enters a new question and clicks Add Question	6a	The system adds this question to the form
5b	User selects a question and clicks Delete Question	6b	The system deletes this question from the form
5c	User edits a question text and clicks Apply Changes	6c	The system will save any changes that the user made to the question.
		7	The system displays the form editing page

Exceptions:
- If there are no forms for this department or the user is not associated with a department
  - The list is empty.

Use Cases Utilized: none
Notes and Issues: none

Use Case ID: UC-9
Use Case Name: SendStudentEmails
Primary Actor: OCECS Representative
Goal: To send reminder emails to students

Pre-Conditions:
- The OCECS representative is logged on to the system.

Post-Conditions:
- Emails are scheduled to be sent

Main Scenario:

Step	Actor Action	Step	System Reaction
1	User selects the Send Email page	2	The system displays a message telling the user who will be emailed
3	User selects to send the emails	4	The system schedules the emails to be sent at a later time
		5	The system sends the emails at the scheduled time

Exceptions:
- There is a mail server problem
  - The emails will be scheduled to be sent later

Use Cases Utilized: none
Notes and Issues: none

Use Case ID: UC-10
Use Case Name: DepartmentWaiveCoop
Primary Actor: OCECS Representative, Department Representative
Goal: To waive a co-op for a student

Pre-Conditions:
- The user is logged on to the system.
- The user has permission to waive co-ops

Post-Conditions:
- Waived co-op is stored in the database

Main Scenario:

Step	Actor Action	Step	System Reaction
1	User selects to waive a co-op	2	The system adds a default “waived” form as that student’s evaluation and work report
		3	The system adds the name of the person authorizing the waiver as the co-op contact name
		4	The main page is displayed

Exceptions:
- None

Use Cases Utilized: none
Notes and Issues: none

Use Case ID: UC-11
Use Case Name: DepartmentMailStudents
Primary Actor: Department Representative
Goal: To send an email to students on co-op

Pre-Conditions:

- The user is logged on to the system.
- The user has permission to send mail

Post-Conditions:

- Emails are scheduled to be sent

Main Scenario:

Step	Actor Action	Step	System Reaction
1	User selects to send mail to students	2	The system allows the user to input a message
3	User inputs the email message and selects to send the email	4	The system schedules the emails to be sent out at a later time
		5	The system sends the emails

Exceptions:

- There is a mail server problem
  - The emails will be scheduled to be sent at another time
- The user is not associated with a department
  - The user will not be allowed to send email

Use Cases Utilized: none
Notes and Issues: none

13.3 Glossary

Name	Aliases	Definition
ANT		A build system maintained by the Apache Foundation that provides a way to manage building and deploying projects

Lightweight Directory Access Protocol	LDAP	An online directory service. RIT has an LDAP entry for every DCE account.
Co-op Block		A period of 10 weeks where a student is working on a cooperative education assignment. A block often lines up with a quarter' schedule.
Cooperative Education	Co-op	An assignment in which a student works in his/her field of study for a real-world employer.
Distributed Computing Environment Account	DCE account	An RIT computer account, which grants access to online RIT services.
Employer Evaluation	Employer Evaluation Form	An evaluation of a student's performance while on co-op, completed by the student's employer.
Family Educational Rights and Privacy Act	FERPA	A federal law that protects the privacy of student education records. (Title 20 United States Code § 1232g; 34 CFR Part 99)
Information Technology Services	ITS	The department that will be housing and maintaining the system.
Likart-Scale Response		A "bubble" response, in which the evaluator chooses a rating from a given scale.
Office of Cooperative Education and Career Services	Co-op Office, OCECS	A department at RIT that is responsible for managing the RIT co-op program.
Online Co-op Evaluation System		The system being developed to replace the current method of processing co-op evaluation forms.
Rochester Institute of Technology	RIT	
Student Work Report		The evaluation that a student fills out at the end of a co-op evaluating how well they think they did and how much they learned
Text Response		A form field where an employer may freely enter any amount of text.