Software Requirement Specification (SRS) for Co-op Support Application

1. Introduction

1.1 Purpose:

The purpose of the co-op support application is to help streamline co-op applications from students and allow the co-op coordinator to review and judge such applications. The software should further double as a portal for future/current co-op students to submit their report for their work term and allow their employer to evaluate their performance.

The purpose of this document is to outline the features of this application, providing the developers a better understanding of what needs to be done to further serve and validate the features of the system for the client.

1.2 Scope:

We will now compose a list of features that will be integrated into the system (in scope) as well as features that will not be integrated (out of scope).

In Scope:

- Creation of accounts for users, admin and work supervisors.
- Maintenance of applications by co-op coordinators.
- Submitting reports upon completion of a work term.
- Employer evaluation for work term performance.
- Sending a reminder via email to co-op students that have not submitted work term reports on time.

Out of Scope:

- Applying for jobs.
- Reminding or collecting co-op fees.

1.3 Definitions & Acronyms:

Acronyms:

CSA: Co-op Support Application

SRS: Software Requirement Specification

Definitions:

Admin: Upholds and maintains the functionality of the application.

User: A student who wishes to be a part of or is a part of the co-op program.

Co-op Coordinator: The directing body of the co-op program.

Supervisor: An employer overseeing a student within the co-op program.

Application: An entry containing a students full name, student id, and email address.

Provisional Acceptance: A state where a co-op coordinator has accepted a users

application, but they still need a final confirmation at a later deadline up to the co-op coordinators discretion.

Accepted: A state where a user has been fully accepted into the co-op program.

1.4 References:

Appendix A: User Screens

1.5 Overview:

To further define the scope of the report, an overview of the following will be given: In section 2, a description of the use cases that the application will support will be explained. This will further allow us to outline the constraints and the different types of users who will be using our application. Assumptions and dependencies will also be touched upon in the process of designing the application. In section 3, description of the functional requirements will be provided. In section 4, the potential features of the software will be discussed.

2. Overall Description

2.1 Product Perspective:

The goal of the co-op support application is to make applying for co-op, submitting reports and reviewing applications easier for the students, the coordinators and the work supervisors, respectively. We will implement this in a straightforward manner, in an effort to maximize efficiency of the co-op program.

2.2 Product Functions:

Following is a list of the use cases that the co-op support application should follow:

Class of use cases	Use Cases	Description of use cases
Use case related to the system	Maintain the system	Home screen of the administrator who keeps track of the accounts and applications.
Use cases related to the system access	Login to the system	User can log in to their account.
	Change password	User changes their password.
Use cases related to accounts	Create account	Create an account for a new user.

	Edit account	Users are provided with such accessibility to update personal information and change username if needed.
	Delete account	Remove a user's account from the system.
	Suspend account	Suspends a user's account.
Use cases related to applications	Create application	User creates an application.
	Review application	Co-op coordinator reviews a users application and can provisionally accept them or fully accept them.
	Reject application	Co-op coordinator rejects a user's application.
	Accept application	Co-op coordinator accepts a user's application.
	Submit a report	User submits a work report of their work term.
	Submit an evaluation	Work supervisor submits an evaluation of the user.
Use case related to alerts	Send a reminder email	User receives emails regarding the deadline of their application.

2.3 User Characteristics:

a. The user must be enrolled within their respective university.

2.4 Principal Actors:

There are five principal actors for the co-op support application, the "admin", the "user", the "co-op coordinators", the "work supervisors" and the "system".

2.5 General Constraints:

The system can only be accessed online.

2.6 Assumptions and Dependencies:

a. The device must have internet access.
b. Admin can only make functional changes at the co-op director's discretion.
c. Work term reports are only for fully accepted users regardless of placement.
d. Workplace evaluations can only be done on users employed/overseen by the supervisor(s).

3. Specific Requirements:

3.1 Functional Requirements:

The following will provide a list of use cases describing the functional requirements of each.

Use Case 1: Maintain the System

Primary Actor: Administrator

Pre-Condition: N/A

Basic Flow:

1. The admin enters their username and password into the system.
2. System authenticates the pair.
3. The home page is displayed.
    a. A list of all users.
        i. Their individual application statuses.
        ii. If they have their work term report, if applicable and highlighted if they are late.
    b. A list of all supervisors, and co-op coordinators accounts.

Alternative Flow:

2(a). Authentication fails.

2(a) 1.The system prompts them to enter their login information again.

Post Condition: The admin is able to login to the system.

Use Case 2: Log into the System

Primary Actor: User

Pre-Condition: Must have created an account with the system previously.

Basic Flow:

1. User is prompted for their username and password credentials.
2. System authenticates the information provided.
3. Home page is displayed (different for each type of user).
    a. Co-op application is displayed for the respective student.
    b. All applications are displayed for the co-op coordinator(s).

c. Evaluation forms will be displayed for the Work supervisor.

Alternative Flow:

2(a). Authentication had failed.

2(a) 1. System informs the user, incorrect login information has been entered.

2(a) 2. Allow the user to re-enter the information. Three chances are allocated.

Post Condition: User logs in successfully into the application.

Use Case 3: Change Password

Primary Actor: User

Pre-Condition: The user is logged into their account.

Basic Flow:

1. The user clicks on the "Change Password" button.
2. The user is prompted for their current password and their new, desired password.
3. User enters the following fields and hits "Confirm".

Alternative Flow

3(a). The user enters the wrong password in the "Current Password" Field

3(a) 1. The system outputs an error messaging claiming that the current password is incorrect.

3(a) 2. Allows the user to re-enter their current password and desired password.

3(b). User cancels the password change.

3(b) 1.User hits the "Cancel" button.

3(b) 2. None of the changes are saved, and the user is redirected to the previous screen.

Post Condition: The user successfully changes their password. Their new password is now required when signing in.

Use Case 4: Create Account

Primary Actor: System

Pre-Condition: N/A

Basic Flow:

1. A new user is prompted by the system for the following details which are then inputted into the system:
    a. User name
    b. Password

    c. A group of checkmark boxes asking what user they are:
        i. Student
        ii. Supervisor
            1. If a supervisor is selected, the user should fill out a text field with which company they are associated with.
2. User clicks the "Confirm Account" button.
3. System saves these credentials so users can access the account in the future.

Alternative Flow:

1(c). User who click supervisor but leave the text field blank

1(c). 1. System would change the text field red indicating that the field should be filled in.

1(c). 2. User cannot click the "Confirm Account" button to finish account creation.

Post Condition: Successful creation of a new account for the user or supervisor.

Use Case 5: Edit Account

Primary Actor: User

Pre-Condition: The user is logged on to their account.

Basic Flow:

1. The user clicks on the "Edit Account" button.
2. The account information (address, phone number, and other contact info) is listed in textboxes.
3. These fields can be changed by the user.
4. Once the user is satisfied with the changes, they hit the "Confirm Changes" button.

Alternative Flow:

4(a). User exits the screen without clicking "Confirm Changes".

4(a). 1. Screen prompts user whether they are sure that they want to cancel, and that any changes will not be saved.

4(a) 2. If user cancels this message, they can continue changing their account information.

4(a) 3, If user accepts this message, they will be redirected back to the main page.

4(b). User enters invalid information (ie. invalid phone number or postal code).

4(b). 1. Invalid information fields are highlighted, informs user that what they have entered is invalid.

4(b) 2. User is unable to hit “Confirm Changes” button.

Post Condition: The user’s account information is updated.

Use Case 6: Delete Account

Primary Actor: Administrator

Pre-Condition: Admin must be logged into the system.

Basic Flow:

1. Admin finds account to be removed in the list of existing accounts.
2. Admin clicks “Remove” beside this account.
3. System prompts admin to review the selection and make sure they want to delete the account.
4. Admin confirms termination of the account.

Alternative Flow:

4(a). Admin denies the termination of the account.

4(a). 1. System sends admin back to the main page.

Post Condition: Account is terminated successfully.

Use Case 7: Suspend Account

Primary Actor: Administrator

Pre-Condition: Admin is logged into the system.

Basic Flow:

1. Admin goes through the Users currently who have an account.
2. Admin selects the “suspend” button beside the name of the user.
3. System prompts the admin to review the correct account to be suspended.
4. Admin confirms the following action.

Alternative Flow:

4(a). Admin denies the selection.

4(a) 1.Admin is sent back to the main page.

Post Condition: The user's account is successfully suspended.

Use Case 8: Create Application

Primary Actor: User

Pre-Condition: User is logged into their account.

Basic Flow:

1. User clicks on the “Create Application” button.

2. The system prompts the user for their credentials:
    a. Full name
    b. Student number
    c. Email address
3. The user submits the following fields and clicks on the “Submit Application for Review” button.

Alternative Flow: N/A

Post Condition: User has successfully submitted an application to the co-op program.

Use Case 9: Review Application

Primary Actor: Co-op Director

Pre-Condition: Co-op director must be logged into the system

Basic Flow:

1. Director opens the list of active applications
2. Locates the desired application
3. Clicks the “view application” button

Alternative Flow: N/A

Post Condition: Director has successfully reviewed a co-op application

Use Case 10: Reject Application

Primary Actor: Co-op Director

Pre-Condition: Co-op direction must be logged into the system and viewing a user’s application

Basic Flow:

1. Clicks the “Reject Application” button located beside the user’s application
2. Co-op director can leave an optional comment on why the application was rejected
3. Pop-up box opens up and prompts director to confirm this action
4. Director clicks “confirm”

Alternative Flow:

4.(a) director clicks “cancel” when prompted to confirm

4(a) 1. Pop-up box disappears

4(a) 2. Director is sent back to list of user applications

Post Condition: The user’s application has been rejected by the co-op director

Use Case 11: Accept Application

Primary Actor: Co-op Director

Pre-Condition: Co-op direction is viewing a user's application

Basic Flow:

1. Clicks the "Accept Application" button located beside the user's application
2. Pop-up box opens up and prompts director to confirm this action
3. Director clicks "confirm"

Alternative Flow:

3.(a) director clicks "cancel" when prompted to confirm

3(a) 1. Pop-up box disappears

3(a) 2. Director is sent back to list of user applications

Post Condition: The user's application has been accepted by the co-op director

Use Case 12: Submit Report

Primary Actor: User

Pre-Condition: User must be logged into the system

Basic Flow:

1. User clicks "submit report" button
2. System asks user to choose either "upload from computer" or "drag and drop file(s)"
3. User selects "upload from computer"
4. System opens users file explorer application
5. User finds their report and clicks "submit"
6. Pop-up box prompts user to confirm their submission
7. User selects "confirm"

Alternative Flow:

3(a). User selects "drag and drop file(s)"

3(a) 1. System provides a space for the user to drop their file

3(a) 2. User opens their file explorer, then drags and drops desired file onto the given space

3(a) 3. Pop-up box appears asking to confirm this submission

3(a) 4. User selects "confirm"

7(a). User selects "cancel"

7(a) 1. Pop-up box disappears

7(a) 2. User is sent back to main page

Post Condition: User has successfully submitted a report

Use Case 13: Submit Evaluation

Primary Actor: Work supervisor

Pre-Condition: Work supervisor must be logged into the system

Basic Flow:

1. Work supervisor clicks “submit evaluation” button
2. System prompts choice between either “upload from computer” or “drag and drop file(s)”
3. Work supervisor selects “upload from computer”
4. System opens their file explorer application
5. Work supervisor locates their report and clicks “submit”
6. Pop-up box prompts confirmation of submission
7. Work supervisor selects “confirm”

Alternative Flow:

3(a). Work supervisor selects “drag and drop file(s)”

3(a) 1. System provides a space for the work supervisor to drop their file

3(a) 2. Work supervisor opens their file explorer, then drags and drops desired file onto the given space

3(a) 3. Pop-up box appears asking to confirm this submission

3(a) 4. Work supervisor selects “confirm”

7(a). Work supervisor selects “cancel”

7(a) 1. Pop-up box disappears

7(a) 2. Work supervisor is sent back to main page

Post Condition: Work supervisor has successfully submitted an evaluation

Use Case 14: Send Email Reminders

Primary Actor: Administrator

Pre-Condition: Admin must be logged into the system

Basic Flow:

1. Admin locates the account that requires an email in the Users List.
2. Admin clicks “Send Email” button beside this account
3. System prompts admin to compose the email via a pop-up box.
4. Admin finishes writing the email.
5. Admin clicks the “send” button

Alternative Flow:

____3(a). Admin decides to not write the email

3(a) 1. Admin clicks the cancel button

3(a) 2. The pop-up box disappears and screen only displays the Users List

Post Condition: Admin has successfully sent email reminder(s)

3.3 Performance Requirements:

a. Any computer with a minimum of 2 GB of memory.
b. Any computer with a minimum of 250 GB storage.
c. System must be compatible with all operations systems (E.g. Windows, Linux, macOS etc.)
d. System must be accessible to multiple people at one time.
e. Response time for most actions can vary from 2 seconds to several minutes. This time relies heavily on the primary actor and how quickly they are performing these actions.

3.3 Complexity Calculations

Actor Complexity

Actor	**Complexity**
System	Simple
User	Complex
Co-op Director	Complex
Admin	Complex

Use Case Complexity

Use Case	**Complexity**
Maintain the system	Complex
Login to the system	Average
Change password	Average
Create account	Simple
Edit account	Complex

Delete account	Average
Suspend account	Average
Create application	Average
Review application	Average
Reject application	Simple
Accept application	Simple
Submit a report	Complex
Submit an evaluation	Complex
Send a reminder email	Complex

UAW Calculations

Complexity	**Description**	**Weight**	**Number of Actors**	**Total**
Simple	Another system with a defined API	5	1	5
Average	Another person that interacts through a protocol or a text based interface	10	0	0
Complex	Person interacting through a GUI	15	3	45
			Total:	50

UUCW Calculations

Complexity	**Description**	**Weight**	**Number of Use Cases**	**Total**
Simple	3 or less transactions	5	3	15
Average	4-7 transactions	10	6	60
Complex	7 or more transactions	15	5	75
			Total:	150

UUCP Complexity

150 (UUCW) + 50 (UAW) = 200 (UUCP)

3.4 Design Constraints:

1. **Security:**
    a. Login credentials are necessary as it is proof of authentication and authorization.
    b. Each individual's information must be secured and not accessed by any other user. Two levels of security must be implemented: general user and administrative. Administrative level should only have access to things such as who has submitted an application and who is currently logged in.

2. **Fault tolerance:** If the system crashes or an action fails, the user account information should not be lost, damaged or corrupted. If an action is being processed when a crash occurs, the system should prioritize the security of the account before saving the state of the action.

3.5 External Interface Requirements:

The CSA system should have two different interfaces corresponding to different security clearance levels. User friendliness should be prioritized by both interfaces and the system must make it clear which security level the user is currently in. Available actions vary by interface and the type of user logged into the system. For example, an employer can submit evaluations while a student cannot. The goal of the system interface is to help the user navigate through the application efficiently.

Administrative clearance level interface must be able to:

- View all the users of the system and their account information, type of account (student, supervisor, coordinator, etc.) and account status (active or suspended).
- Edit, suspend and delete user accounts.
- View submitted reports, evaluations and applications and their status.
- Accept or reject co-op applications.

General user clearance level interface interface must be able to:

- View and edit personal account information.
- Submit co-op application or work term report.
- Track the status of their submitted application.

4. Future Extensions:

Future extensions for the Co-op Support Application may include:

- ➢ Accepting co-op fees through this application.
- ➢ Providing an option for users to receive notifications about their ongoing applications.
- ➢ Creating a mobile application to make the system more accessible for mobile devices.
- ➢ For users that have been accepted into co-op:
    - Allow them to browse job opportunities.
    - Supplying additional resources to help students with their job applications. (I.e tips, contact info for advisors, etc.)

5. Appendix:

5.1 Appendix A - User Screens:

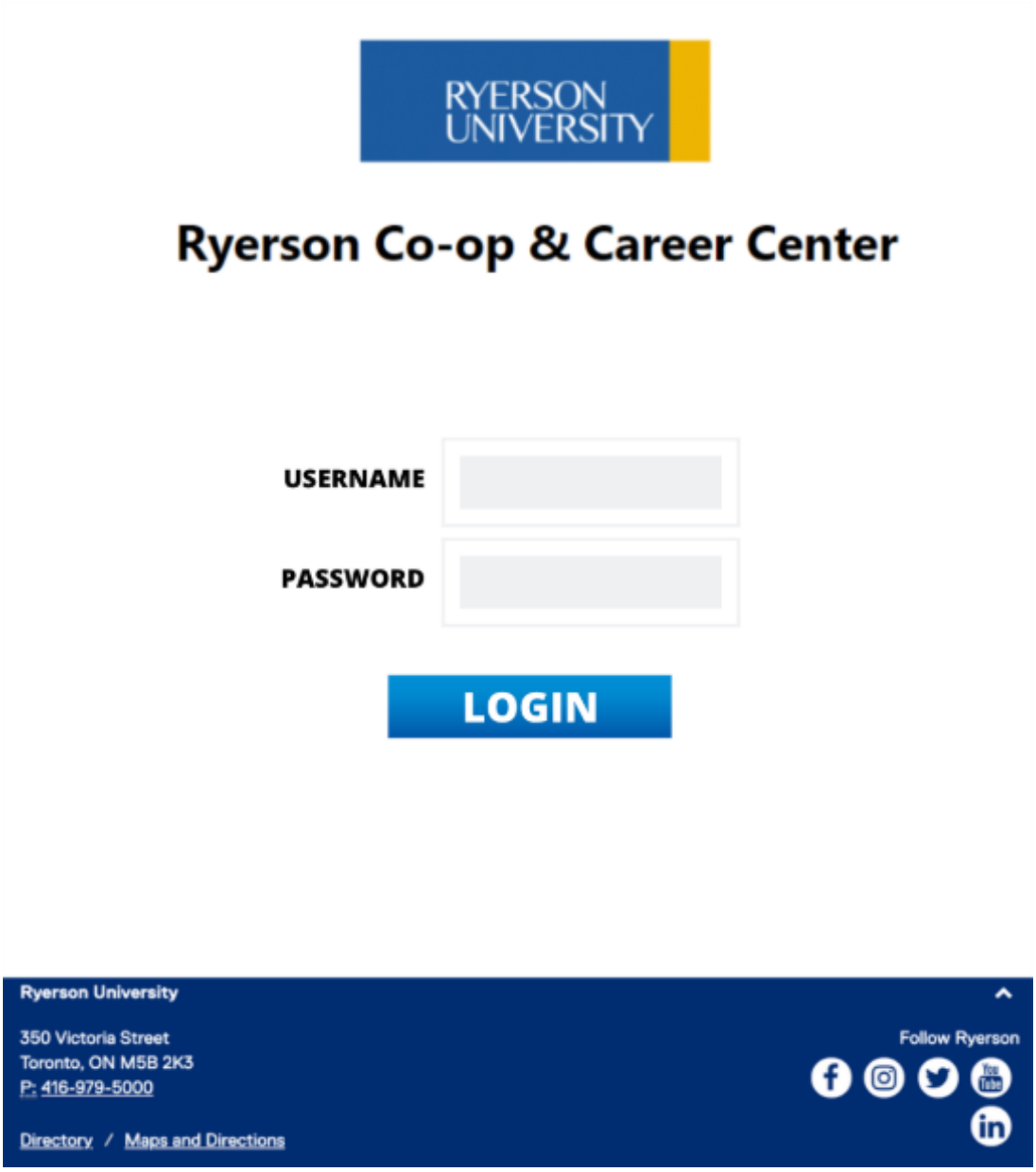

Figure 1: Login page

RYERSON UNIVERSITY

Ryerson Co-op & Career Center

Home | View Application | Submit Report | Work Term History | Account Settings

News:

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Click here for more

Upcoming Dates:

Date	Event
Monday, Febuary 17 2020	Applications Due
Sunday, March 1 2020	Fall 2019 Work Term Reports Due

Figure 2: Home page once user logs in

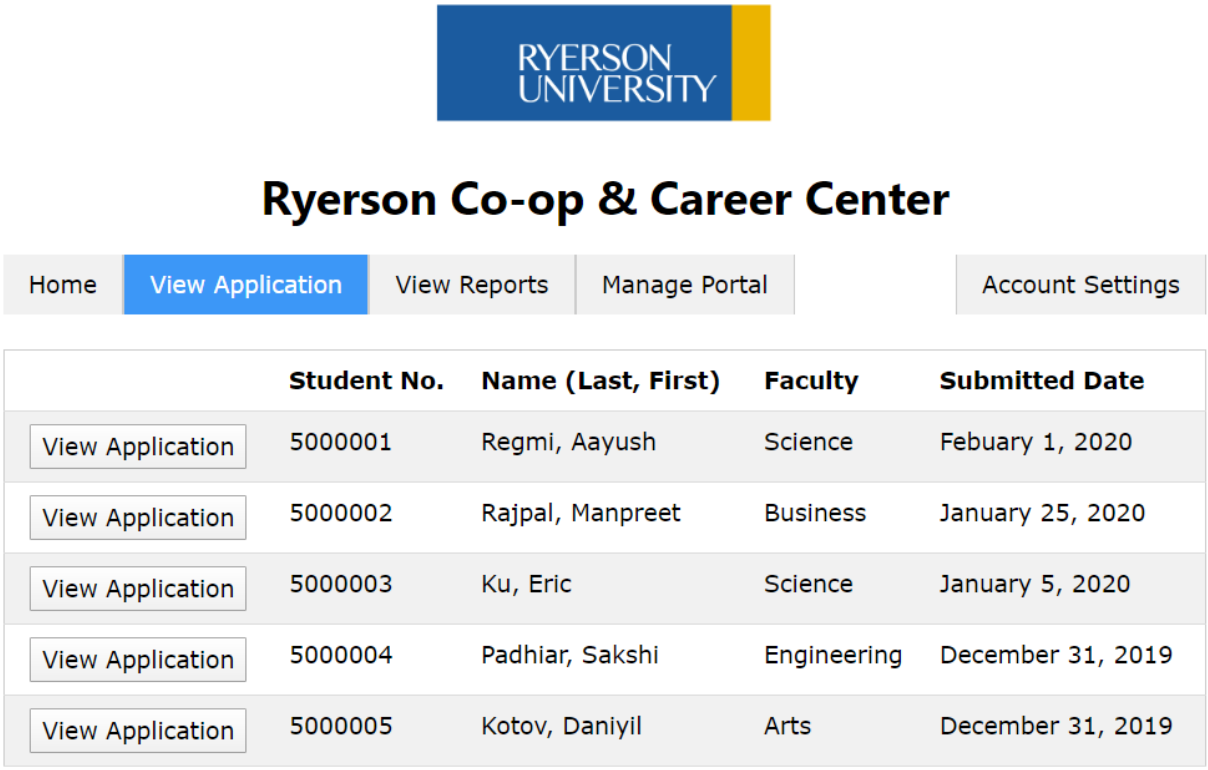

Figure 3: Co-op directors page for viewing application