Software Requirements Specification

COSC 4P02

Version 1.0

Due: Sunday 7th Feb at 23:55

R8Scholar

Team Leader:
Twino Puthiakunnel / mp13ko@brocku.ca / 5564182 / mp13ko
Secondary Team Leader:
Seth Shickluna / ss16wn@brocku.ca / 6217558 / SethShickluna
Erikas Klimusinas / ek15kg@brocku.ca / 5903547 / ErikasK
Grant Nike / gn17az@brocku.ca / 6349302 / GrantNike
James Sargant / js17sy@brocku.ca / 6380356 / shorinbonsai
Logan Bell / lb16tp@brocku.ca / 6047211 / loganbe11
Luciano Ugalde / lu16xx@brocku.ca / 6102545 / lu16xx
Munashe Masango / mm16rh@brocku.ca / 6204911 / munashemasango

Table of Contents

Note* to add things to the Table of contents automatically, highlight the heading, then go to Format>Paragraph styles then pick the type of heading it is.

e.g. **1.0. Introduction** is type Heading 1
1.1. Purpose is type Heading 2
1.1.1 Example is type Heading 3

Click on the Table of Contents and hit the refresh button to the left of the Table of Contents to refresh with any newly added Headings

SRS Table of Contents Planning Template //Subject to change, please add things

1. Introduction

1.1 Purpose of this document

1.2 Scope of this document

1.3 Overview

1.4 Business Context // not sure if this one is needed

2. General Description

2.1 Product Functions

2.2 Similar System Information // not sure if this one is needed

2.3 User Characteristics

2.4 User Problem Statement // not sure if this one is needed

2.5 User Objectives

2.6 General Constraints

3. Requirements

3.1 User Interface Requirements

3.2 System Requirements

3.2.1 System Functions

3.2.2 System Services

3.2.3 Operational Constraints

3.3 Stakeholder Requirements

3.4 Functional Requirements

3.4.1

3.5 Non-Functional Requirements

3.5.1 Product Requirements

3.5.2 Organizational Requirements

3.5.3 External Requirements

4. Interface Requirements

4.1 User Interfaces

4.2 Hardware Interfaces // not sure if this one is needed

4.3 Communications Interfaces // not sure if this one is needed

4.4 Software Interfaces // not sure if this one is needed, maybe mobile?

5. Performance Requirements

6. Other non-functional attributes

6.1 Security

6.3 Reliability

6.4 Maintainability

6.5 Portability

6.6 Extensibility

6.7 Reusability

1.0. Introduction

1.1 Purpose

COSC 4P02 R8Scholar Requirements Elicitation:

Stakeholders:

- Us (Developers)
- Prof Naser
- Users (brock students / people with a brocku email) - Anonymous
- Testers (Perhaps our non COSC brock friends?)
- Potentially the people being rated on the site

Expectation of the system:

- Online platform to rate scholarly aspects of Brock University
- Anyone should be able to view all ratings on the site
- Verified brock students should be able to leave ratings on the site
    - Signup / Login
        - Verify Brock Email
- Needs to display a page for each department
    - Departments can list staff and courses
        - All of these should be dynamically generated, probably based off database entries
    - Search feature on front page?
-
-

Use Cases:

- Create a review for professors, courses, departments, other
    - Comments
    - Rating
    - Option to add tags about professor characteristics
        - Hard marker
        - Timely
        - etc
- Reply to Review
- Create new student account
- Sign in to student account
- Students should be able to search up professors or courses
- General forum thread for each topic of discussion
- Viewing professor profile page
- Add Professor/Course/other to site
- Modify Professor/Courses/Other

- Search Feature

System Architecture:

Django based

- Model - Database structure
- View - Site control
- Template - HTML templating
-

Risk Factors of the system:

- Leaked information
  - industry standard password encryption
  - AWS has security features built into the servers they host
- Site crashing?
- Unprofessional comments?
  - Run comments through a profanity filter before posting?
  - Report function

Diagrams required:

Use Case Diagrams:

3.1.5-3.1.14 - Luc

3.1.1-3.1.4 + 3.1.15-3.1.17 - Grant

Class Diagrams TBD [Sunday ~2PM]

Context model/Activity Diagrams (Email verification - scraping) [Sunday ~2PM]

Sequence Diagrams

State Diagrams (viewer/student/teacher/admin states)

// This is where I'm placing the basic list of functional and non-functional requirements in order to find them so I can edit and make detailed tables for them. Please feel free to add the ones you think of. //-mm16rh

3.0. Requirements

3.1 User Interface Requirements

3.2 System Requirements

3.2.1 System Functions

3.2.2 System Services

3.2.3 Operational Constraints

3.3 Stakeholder Requirements

// Will be filled after talking to Professor Naser probably

3.4 Functional Requirements

- Users can sign up for the website =3.1.1
- Users can log in (Brock email) (authentication)=3.1.2
- Users can log out = 3.1.3
- Users can edit their profile information = 3.1.4
- Users can rate an entity = 3.1.5
- Users can write a review on an entity =3.1.6
- Users can edit their ratings and reviews =3.1.7

• Users can search the site for an entity =3.1.8

- Users can comment on existing reviews 3.1.9
- Users can edit their comments on existing reviews 3.1.10
- Users can view the comments for any review 3.1.x

• Users can view content by category (type of entity) = 3.1.11

• Users should be able to contact the system admin as well as see the about information for the website 3.1.12

• Users can add to forum section (attached to an entity) ...3.1.13

- Users can a comment to a forum post 3.1.14

Entities can:

• Respond to a review = 3.1.15

• view all reviews (profile level) = 3.1.16

• view all ratings (profile level) = 3.1.1x

• search for reviews = 3.1.17

3.5 Non-Functional Requirements

3.5.1 Product Requirements

Requirements which specify that the delivered product must behave in a particular way e.g. execution speed, reliability, etc.

, Efficiency(Performance, Space), Dependability, Security

Usability

- There should be an option for colourblind or other colour scheme themes in order to have the site be usable by as many different types of users to be given access.
-

Efficiency

- The search should be efficient (auto completing or querying as they type)

Security

- The authentication is only for brocku.ca domain accounts and accounts should be validated as real Brock accounts
  - User account information should be handled and stored securely(i.e hash passwords before storing)

Dependability

- Only one account for each email address.

3.5.2 Organizational Requirements

Requirements which are a consequence of organizational policies and procedures e.g. process standards used, implementation requirements, etc.

Operational, Development

Operational
- Site must update courses and professors as they are added
- Must have the ability for a moderator to remove comments
-

3.5.3 External Requirements

Requirements which arise from factors which are external
to the system and its development process e.g. interoperability requirements, legislative requirements, etc.

Regulatory, Ethical, Safety/security

// Will be filling this in more after talking with Professor Naser

*3.1
Tables:

3.1.1

Use Case Name	Sign Up / Create Profile
XRef	.
Trigger	The user selects the “sign up” on the website
Precondition	The user has not created a profile for their email address
Basic Path	1. The system opens a form to enter information 2. The user will enter the correct format of data into the form 3. The system confirms the information

	with the database and the user 4. The profile is created and the user redirected to the profile page
Alternative Path	None noted as of now
Postcondition	The database has been updated
Exception Paths	If the user has already created a Profile using that given email address then the use case abandons
Other (Notes)	This can only be used for each email address once

3.1.2

Use Case Name	Log in to profile
XRef	.
Trigger	The user selects the “log in” on the website
Precondition	The user has created a profile previously using a defined email address
Basic Path	1. The system opens a log in dialogue 2. The user inputs the correct format of data (text) 3. The database authenticates whether that user is allowed access 4. The user is redirected to the
Alternative Path	There is the possibility of being directed to the login via a direct link
Postcondition	The user is redirected and the log are updated
Exception Paths	If the user inputs invalid information there will be an error and it will allow you to re-enter the text.
Other (Notes)	

3.1.3

Use Case Name	Log out of profile
XRef	.

Trigger	The user selects the log out option on the website
Precondition	The user was already logged in and accepted into the database.
Basic Path	1. The system prompts the user to make sure they want to log out 2. The system will redirect the user to the homepage of the website
Alternative Path	None noted so far
Postcondition	The user is no longer authorised within the database.
Exception Paths	The user decides against logging out of the website and thus the use case is abandoned.
Other (Notes)	

3.1.4

Use Case Name	Edit a profile
XRef	.
Trigger	The user selects the Edit feature within the profile.
Precondition	The user must be authorised, be viewing the profile page they created.
Basic Path	1. The profile information form is presented in an editable view 2. The user edits the desired fields in order make changes 3. The system confirms before making any changes to the profile.
Alternative Path	None noted so far.
Postcondition	The database is updated with the new information from the form.
Exception Paths	If the user updates the information with invalid content it will prompt them to fix it. If the user cancels the changes, there are no new changes and the use case is abandoned.

Other (Notes)	

3.1.5

Use Case Name	Rate an entity
XRef	.
Trigger	The user enters a value into the rating fields
Precondition	The user must be authenticated in order to be able to rate an entity.
Basic Path	1. The user selects their numeric rating of the entity in question 2. The system prompts the user to confirm the score they have put in. 3. The system shows the user the rating they just input
Alternative Path	None noted
Postcondition	The database is updated and thus calculates the new average score for said entity
Exception Paths	If the user does not confirm their entry the system will abandon the rating process and revert the database to the previous state.
Other (Notes)	Rating an entity with 0 or -1 means it will essentially be removed

3.1.6

Use Case Name	Write a review
XRef	.
Trigger	The user selects the review section of the entity's profile
Precondition	The user should be authorised as well as having previously rated said entity
Basic Path	1. The user inputs the text into the editing field 2. The system confirms the text entered is complete. 3. The system publishes the review to

	the profile of the given entity
Alternative Path	None noted
Postcondition	The database is updated with the new review
Exception Paths	If the user chooses not to publish said review the system will leave the editing view and abandon the use case
Other (Notes)	

3.1.7

Use Case Name	Edit review
XRef	.
Trigger	The user selects the edit function on the review.
Precondition	The user must be authorised as well as being a review that is created by that user.
Basic Path	1. The system makes sure the input is of the correct type (text for a review or numeric for ratings) 2. The system updates the entities profile 3. The system shows the user the newly edited review
Alternative Path	None noted so far
Postcondition	The database is updated with the new information we get from the review section.
Exception Paths	If the user chooses not to publish said edit the system will leave the editing view and the use case abandoned
Other (Notes)	This also includes the ability to remove a review

3.1.8

Use Case Name	Search for an entity
XRef	.
Trigger	The user selects the search field in the

	website
Precondition	There is a non-empty database of entities and thus can be queried
Basic Path	1. The system reads the input from the user 2. The database queries for the given conditions of the search parameter 3. The results (entities) are displayed by the system 4. The user picks an entity from that list 5. The system redirects the user to the corresponding profile
Alternative Path	None noted
Postcondition	None noted
Exception Paths	If the user inputs a search term that does not correspond to anything we have stored in the database, the user will be prompted to change the search term
Other (Notes)	

3.1.9

Use Case Name	Write Comment
XRef	.
Trigger	The user selects the comment section below a review that has been posted
Precondition	The user must be authorised and there must exist a comment to use.
Basic Path	1. The system awaits the input from the user of correct type (text) 2. The system confirms if that text if final 3. The system then shows the user the comment they made as well as the original review
Alternative Path	None noted
Postcondition	The database is updated with the new comment to the given review
Exception Paths	If the user chooses to cancel the comment

	the system should remove the text and revert to the previous view
Other (Notes)	

3.1.10

Use Case Name	Edit Comment
XRef	.
Trigger	The user selects the ‘edit’
Precondition	The user must be logged in and there must be a previously made comment by the same user
Basic Path	1. The system will wait for the user to enter the text 2. The system will confirm the text with the user 3. The comment text will be published replacing the old with the new text
Alternative Path	None noted
Postcondition	The database is updated, changing the previously saved text with the new text.
Exception Paths	If the user wants to, there is the ability to cancel the edit, leaving the old text unchanged in the database.
Other (Notes)	

3.1.11

Use Case Name	View entities by category
XRef	.
Trigger	The user selects the option on the website to view “all” the members of some category.
Precondition	The category of entity (Professor, Course etc) should not be empty within the database
Basic Path	1. The system waits for the user to select the given category of entity. 2. The system will query the database for the given data type

	3. The system displays the entities in that list 4. The system waits for further action from the user
Alternative Path	None noted
Postcondition	None noted
Exception Paths	If there is nothing within the database that matches the given category the use case should abandon and redirect the user back to the previous page (use case).
Other (Notes)	

3.1.12

Use Case Name	Contact System Administrator
XRef	.
Trigger	The user selects the “contact us” section of the website.
Precondition	There are no pre-conditions
Basic Path	1. The system presents the user with a form to contact the administrators of the system 2. There are text fields for the system to accept inputs from the user 3. The systems asks the user to confirm the message they are looking to send 4. The message is sent to the administrators, who can then respond to any issues raised
Alternative Path	The user may also find the form through an external hyperlink provided in some other medium i.e. email
Postcondition	There will be an update to users who are already authenticated, there will be an update to the database with the details of the contact the user has had with the administrators.
Exception Paths	The user can cancel any form of communication and not send through the message, in this case the use-case is

	abandoned and there are no postconditions for registered users.
Other (Notes)	

3.1.13

Use Case Name	Create Forum Post
XRef	.
Trigger	The user selects the community section on and entity’s profile
Precondition	The user should be authorised in order to gain access to the community section of the profile
Basic Path	1. The system waits for the user to input the text into the edit field 2. The system asks the user to confirm their message for the forum post 3. The system publishes the message and then allows for comments and interaction on the forum posts.
Alternative Path	None noted
Postcondition	The database is updated with all the information within the forum post and stored
Exception Paths	If the user cancels their intention of posting to the forum, the use case is abandoned and there are no postconditions
Other (Notes)	

3.1.14

Use Case Name	Comment of Forum Post
XRef	.
Trigger	The user selects the comment function on a previously posted forum post
Precondition	The user must be authenticated within the system and there should also be a forum post already made.

Basic Path	1. The system waits for the user to input the text into the edit field 2. The system asks the user to confirm their comment for thatforum post 3. The system publishes the message and then allows for comments and interaction on the forum comment.
Alternative Path	None noted
Postcondition	The database is updated with the messages written into the comments on that individual
Exception Paths	If the user cancels their intention to comment on the forum post there should be and abandonment of intention and no post conditions
Other (Notes)	

3.1.15

Use Case Name	Respond to Review
XRef	.
Trigger	The Entity selects the reply function on the reviews on the profile.
Precondition	The entity should be authorised and verified to be the responsible party for the review
Basic Path	1. The system waits for the user to input the text into the edit field 2. The system asks the entity to confirm their comment for that forum post 3. The system publishes the message to the review
Alternative Path	None noted
Postcondition	The database is updated with the message added to the review.
Exception Paths	If the entity decides to cancel their intention of replying to a comment the use case is abandoned.
Other (Notes)	

3.1.16

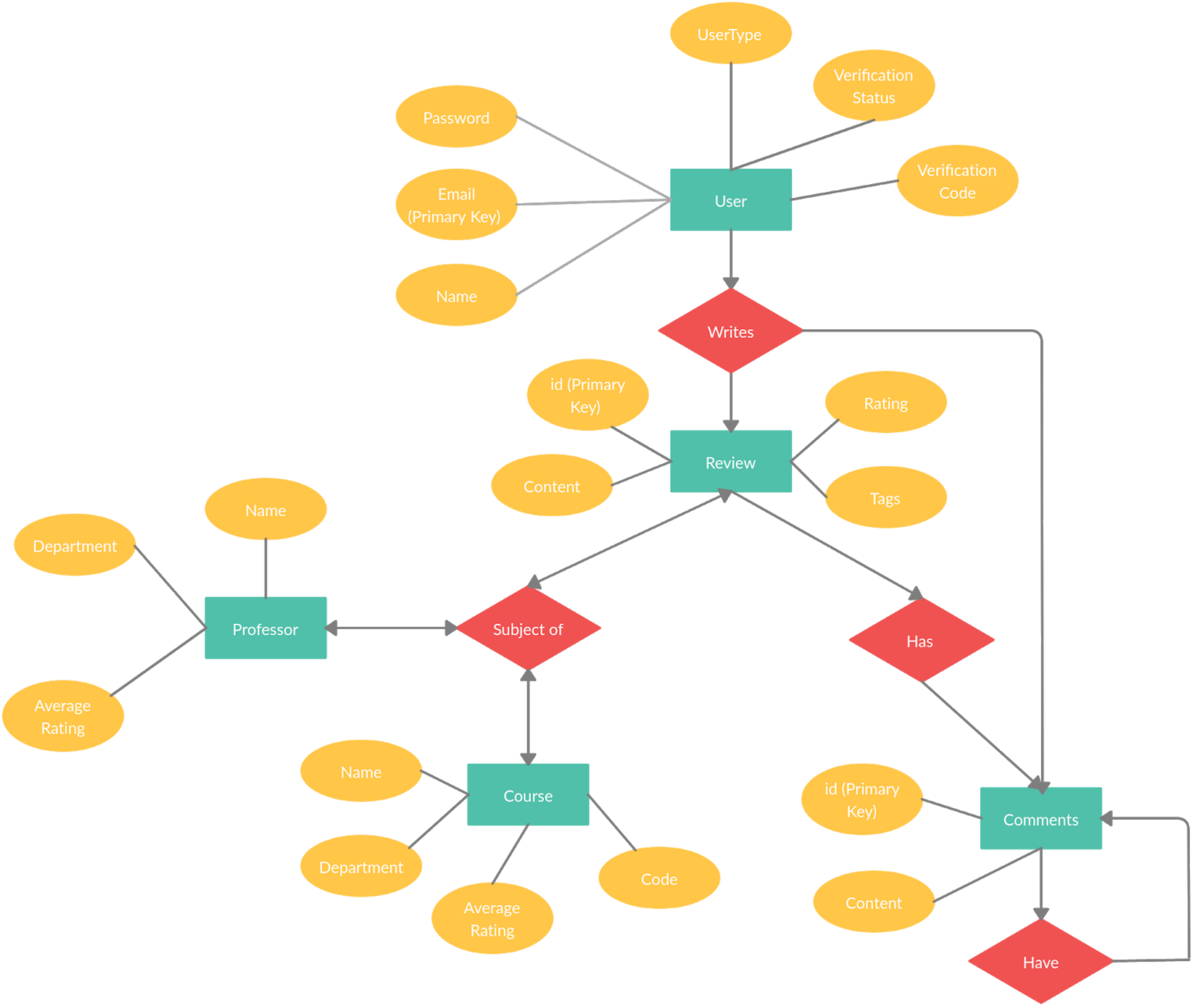
Use Case Name	View all reviews
XRef	.
Trigger	The entity can go to the page on which you can view all the reviews and ratings
Precondition	The entity should be authorised and verified to be the responsible party for the review
Basic Path	1. The system displays a section with all the ratings of that entity 2. The system displays a section with all the review messages written about that entity
Alternative Path	The entity may view all the ratings and reviews by clicking on their score on the profile.
Postcondition	None noted
Exception Paths	If the entity has no ratings and reviews published at that time, the system should notify them of that and abandon the use case.
Other (Notes)	

3.1.17

Use Case Name	Search through reviews
XRef	.
Trigger	The entity selects the search feature on website
Precondition	The entity is already authorised and is on the page where they view all the reviews
Basic Path	1. The entity enters some form of keyword into the search field 2. The system queries the database for that given keyword 3. The system presents the results of the search, listing reviews that match the search term
Alternative Path	None noted
Postcondition	None noted

Exception Paths	If there is nothing matching the search term, the system presents an exception for the user.
Other (Notes)	

4.0. Database Design



User Entity

Data Item	**Type**	**Description**	**Comment**
Email	Text/String	Email of the user	Probably would be the unique identifier/key
Name	Text/String	Name of the user	
Password	Encrypted text/String	Account password	This would be encrypted (only the user themselves would know what it is)
Reviews	Pointer	Review Entity	This would point to each review this user left
UserType	Boolean?	Type of user (dev, student, banned?)	Handles permissions, etc
isVerified	boolean	Used to determine if the user has verified their email	Probably done by sending a code to the provided email
verificationCode	string	Generated upon account creation to verify the given email	Code is emailed to user then they input it Into a first time page

Course Entity

Data Item	**Type**	**Description**	**Comment**
Name	Text/String	Full name of the course (dept + code)	ex: COSC 2P03
department	Text/String	Department in which the course is from	ex: COSC
Code	Text/String	Course code	ex: 2P03
Avg Rating	int/Numerical	The average score	

		this course was rated	
Reviews	Pointer	Review Entity	reviews for course

Professor Entity

Data Item	**Type**	**Description**	**Comment**
Name	Text/String	Full name of the professor	ex: Dave Bockus
department	Text/String	Department in which the course is from	ex: COSC
Avg Rating	int/Numerical	The average score this course was rated	
Reviews	Pointer	Review Entity	reviews for this prof

Review Entity

Data Item	**Type**	**Description**	**Comment**
Id	number	Id number of the review (for book keeping)	
reviewer	User	Points to the user entity who wrote this review	
subject	Professor / Course entity	Points to the subject (professor / course) of the review	
content	Text/String	Text review written by a user	
Rating	int/Numerical	Score this rating gave	
tags	text/String	Pre-defined tags about the course/prof for ease of use	
response	Comment Entity	Points to each comment that has been left on the review	This could be any number >= 0

Comment Entity

Data Item	**Type**	**Description**	**Comment**
Id	number	Id number of the review	
Name	string	Name of commenter	
Content	string	Comment nested in review	
Comment ID	number	Comment number	
Child	Comment entity	Points to all replies to this comment	