CPSC Internship Data Site
Project #:
Requirements Document

Prepared by: Christopher Vasquez, Daniel Schaub, Jorge Contreras

Prepared for:

Date Submitted: February 5, 2019

Project Sponsor: Jennifer Polack , Karen Anewalt

Version: 0.5

Last Updated: February 2, 2019
Creation Date: February 6, 2019

Table of Contents

1. Introduction

1.1. Document Purpose

The purpose of this document is to describe software requirements of an Application completely, accurately and unambiguously in Technology-independent manner. All attempts have been made in using mostly common terminology while describing the requirements in this document. Very minimal and commonly understood Technical terminology is used, as to not confuse the students who will be using this software.

1.2. Intended Audience

The main intended audience for this document are Karen Anewalt and Jennifer Polack the one who proposed this system. This document should be readable by business owners of the proposed system. They must be able to verify that their software requirements have been documented here completely, accurately and unambiguously.

The implementation team would also find the information in this document useful when they need to design a solution that will address these software requirements.

Since the requirements are documented here in Technology-independent manner, the end-users of the system should be able to comprehend the requirements fairly easily from this document.

1.3. Project Background

The Software Requirements stated in this document are a result of previous meetings with the client Jennifer Polack. We discussed in length what capabilities she wanted with the software and we constructed requirements to meet those needs that were feasible for the allotted time for the project.

1.4. Purpose of the Business Requirements

This section describes the purpose of the Business Requirements.

☐ Business requirements for major enhancements to an existing application.

☐ Business requirements for new application development.

☐ Business requirements for replacement application development.

☐ Business requirements for a request for proposals (RFP).

1.5. Business Goals/Objectives to be achieved

The major goals/objectives to be achieved with the implementation of the Software Requirements is to deliver an stable and easy to use website that allows students from anywhere to consult a database of past internships with first hand feedback and additional information from the University of Mary Washington computer science students that worked those specific internships.

1.6. Benefits/Rationale

The major benefits to be achieved with the implementation of this software is that any student will be able to consult a database with feedback from past interns on company contact information, general application deadlines, etc.. Students will be able to search internships by name or through viewing them on a map to be able to select one based on how far or close you would like them to be.

1.7. Stakeholders

The individuals who have a vested interest in this project and whose interests need to be considered throughout the project are;

- Jennifer Polack
- Karen Anewalt

1.8. Dependencies on existing systems

This website will have a dependency with the University of Mary Washington's database of Computer Science students. This is because only University of Mary Washington computer science students who have had an internship can submit feedback on this site, so we need to be able to check the students id number, major, and that they had an eligible internship.

1.9. Assumptions

This section describes major assumptions that were made prior to or during the Business Requirements gathering and documentation.

2. Functional Requirements

This section describes the *Functional requirements* part of the Business Requirements. In Use case approach, the *Functional Requirements* comprises of Actor Profile Specification, Essential Use case diagram and Essential Use case specification in narrative text form. In Oracle Designer approach the *Functional Requirements* comprises of Business Unit Definition Report, Function Hierarchy Diagram and Function Definition Report.

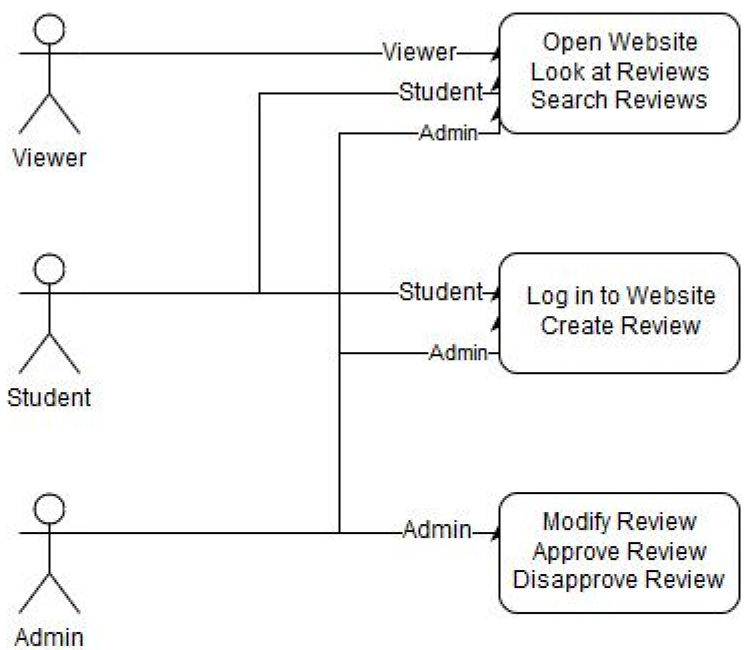
2.1. Actor Profiles Specification

This section describes all the Actors and their profiles within the context of the Software Requirements being documented. An Actor is a person, organization or an external system/sub-system/program that has interactions with the Application. Actors, by definition, are external to the system with which they are having interactions. Actors have goals that are achieved by use cases. Typically, Actors have behaviour and are represented by the roles they play in the use cases. An Actor stimulates the system by providing input and/or receiving something of measurable value from the system.

Actor Name	**Actor Type**	**Access Type needed**		**Comments**
	☐ Stakeholder ☐ Primary Actor ☐ Supporting Actor	☐ Create ☐ Read ☐ Update ☐ Delete	☐ Print ☐ Export ☐ Others	
	☐ Stakeholder ☐ Primary Actor ☐ Supporting Actor	☐ Create ☐ Read ☐ Update ☐ Delete	☐ Print ☐ Export ☐ Others	
	☐ Stakeholder ☐ Primary Actor ☐ Supporting Actor	☐ Create ☐ Read ☐ Update ☐ Delete	☐ Print ☐ Export ☐ Others	
	☐ Stakeholder ☐ Primary Actor ☐ Supporting Actor	☐ Create ☐ Read ☐ Update ☐ Delete	☐ Print ☐ Export ☐ Others	

2.2. Essential Use Case Diagram

This section is applicable only to Use case approach. This section depicts the Software Requirements in the form of Essential Use case diagram. In the Use case approach, the Functional Requirements are decomposed into a number of Essential Use cases. Essential use cases are of primary importance early in a project's requirements/analysis phase. Their purpose is to document the software process that the Application must support without bias to technology and implementation.



2.3. Essential Use Case Specifications

This section is applicable only to Use case approach. This section describes each Essential Use case in narrative text form. A use case typically has one basic course of action and one or more alternate courses of actions. The basic course of action is the main start-to-finish path that the use case will follow, where as the alternate courses represent the infrequently used paths and exceptions, error conditions etc. The complete business logic of a use case such as basic course of action, alternate course of action, pre-condition, post-condition etc is not depicted in the Use case diagram. Rather they are documented in narrative style in use case specifications.

If the number of use cases is less than 15, the Essential Use case specifications in narrative form are included in this SRD in tabular format. Each use case is described in a separate table. If the number of use cases is greater than 15, the Essential Use case specifications in narrative form are attached as a separate document with this SRD.

2.4. Function Hierarchy Diagram

This section depicts the Software Requirements in the form of Function Hierarchy Diagram (FHD).

2.5. Function Definition Report

. This section describes each Software Function in narrative text form.

1. As a user, I want to access the website in a browser
2. As a user, I want to open the page and see reviews and a map with locations of the nearest internship.
    a. See Appendix for Figure 1, Figure 2, and Figure 3
3. As a user, I want to click on a review and have it list the company.
4. As a user, I want to click on a tab to suggest companies that could be chosen from.
5. As a user, I want to see a review tab that's never blank, if none was selected on the map then the review tab should show a list of companies ordered by next nearest location.
6. As a user, I want to have a tab to search for previous reviews. Example search terms could be:
    a. CCSC (a company name)
    b. Manassas (a location)
7. As a user, I want to see the following categories of information:
    a. Company Name
    b. Company Description
    c. Company Contact Information
    d. Date of Contact
    e. Application Deadlines
    f. How they found out about the internship
    g. Did they receive credit for the internship
    h. Paid/Unpaid
    i. Hours per Week/ Salary
    j. Sub Field (i.e. Help Desk, Security, Software Engineering, Testing, etc..)
    k. Languages Used
    l. Software Tools Used / Libraries
    m. Benefits of Completing Internship, Limited 2-3 sentences
    n. Prerequisites to Internship
    o. OPTIONAL: Student Contact Info for future students to get in touch with review leaver for questions
    p. Perks (i.e. provided dorm/food/living wages) Limited 2-3 sentences
8. As a student with a umw login, I want to be able to create an account.
9. As a student with a umw login creating an account, I want to verify my account through my umw email to protect against spoofers.
10. As a student with a umw login, I want to be able to log in to the website with my umw login.
11. As a student with a umw login, I want to be able to submit feedback, i.e. leave a review.
12. As a student with a umw login leaving a review, I want to see the form containing the same categories of information above (Company Name…. Perks) to fill.
13. As Professor Polack, I want to log in with an administrator account with administrator privileges.
14. As an admin, I want to have the ability to modify/edit reviews.
15. As an admin, I want to have the ability to approve a review before publishing it.
16. As an admin, I want to have the ability to disapprove a review before publishing it.
17. As an admin, I want a search function for inappropriate words before reviews are published, and flag the review if found inappropriate (i.e. 'this internship sucks', or other crap (shareholder's words)).
18. As Professor Polack, I want to receive an email notification when a review is flagged.
19. As a flagged review, I wish to not be published until approve by an admin.

3. Non-Functional requirements

This section describes the non-functional requirements part of the Software Requirements. A non-functional requirement is typically a special requirement that is not easily or naturally specified in the text of the use case's or function's event flow. Examples of non-functional requirements include legal and regulatory requirements, application standards, and quality attributes of the system to be built including usability, reliability, performance or supportability requirements.

20. As a user, I don't want to see any rating system (like ratemyprofessor.com)
21. As a user looking at a review tab with a list of companies, I want it to stylistically show only a few companies if the list of companies if too large.
    a. Example: If there are 1000 companies, do not show 1000 companies, show 10.
    b. The number of companies listed is not provided, it is up to what looks best stylistically.
22. As Professor Polack, I want the domain to be the cheapest, preferably free.
23. As Professor Polack, I do not wish the website to have any extra graphic-less page, "and this is the most important."
24. As Professor Polack, I want the review forms to have as many drop-down options as possible as opposed to free-form text boxes.
25. As an admin, I want more drop-downs in review form fill-outs so going through reviews will be easier, faster, and more efficient and less prone to error.
26. As a shareholder, there is no preference in coding languages used, such as SQL or c++, python, etc..
27. As a website, I want to start out purely empty.
    a. As sample data, if I were to be used, I want to be real data collected from students.

3.1. Security Requirements

This section describes the Security requirements part of the Software Requirements.

1. Users with accounts shall be forced to change their password the next time they log in if they have not changed it within the length of time established as "password expiration duration."
2. Users must change the initially assigned login authentication information (password) immediately after the first successful login. The initial password may never be reused.
3. The database system shall ensure that the student id data can be accessed only by authorized users. The database system shall distinguish between authorized and non-authorized users.
4. Users shall not be allowed to update their own internship feedback information after submission, and any such attempt shall be reported to the security administrator.
5. Only holders of UMW id number and eligible internship can enter the information into the database
6. The access permissions for system data may only be changed by the system's data administrator.
7. Passwords shall never be viewable at the point of entry or at any other time.

8. Each unsuccessful attempt by a user to access an item of data shall be recorded on an audit trail.
9. Users shall receive notification of profile changes via preferred communication method of record when profile information is modified.

3.1.0. Authentication

This section describes the Authentication requirements part of the Software Requirements. Authentication is the process of verifying the genuineness of claims that a person/group makes to establish identity/eligibility for access to services.

1. Users that want to make and account must submit their student id numbers, so that they can be authenticated as University of Mary Washington students
2. Users that have their id numbers authenticated must wait for their internship feedback to be authenticated that they indeed had that particular internship before it is released for the public to see.

3.1.1. Authorization and Access Controls

This section describes the Authorization and Access Control requirements part of the Business Requirements at a high-level. Authorization is the process of determining if the person/group, once identified through the "Authentication process", is permitted to have access to certain services. The Authorization and Access Control requirements are best described through a matrix.

1. The only users that are authorized to add to the database system are University of Mary Washington students that have eligible internship experience.
2. The system's data administrator is the only user with authorization to allow existing data entries to be edited and to delete data entries.

3.2. Availability Requirements

This section describes the system availability requirements.

1. The account authentication shall be available during regular university hours of 8 a.m. and 5 p.m..
2. The CIF system shall achieve 99.5% up time.
3. The internship search results shall be available to the customer within 15 seconds for 95% of the times that it is requested. The remaining times it will be available within 20 seconds.
4. The internship database entries and feedback shall be at least 99.0 percent available on weekdays between 6:00 a.m. and 11:00 p.m. local time. The machine shall be at least 99.95 percent available on weekends between 4:00 a.m. and 11:59 p.m. local time.
5. Unless the system is non-operational, the system shall present a user with notification informing them that the system is unavailable.
6. The online registration system shall permit backing up of the registration database while other registration activities are going on. (It is estimated that this requirement reduces duration for which the online registration system would be unavailable to students for maintenance by 15 minutes each calendar day.)

3.3. Usability Requirements

This section describes the system usability requirements. A usability requirement specifies how easy the system must be to use. Usability is a non-functional requirement, because in its essence it doesn't specify parts of the system functionality, but specifies only how that functionality is to be perceived by the user, for instance how easy it must be to learn and operate the system.

1. The new software shall be easy to use by students of the public who may only have limited experience with technology.
2. The search feature shall be able to be used by students of the public without training. A panel representative of at least 95 percent of the general public shall successfully locate the nearest internship to their location on the first try.
3. The software shall be self-explanatory and intuitive such that a search shall be able to produce a result within a minute of the search button being clicked.
4. The new internship database shall be evaluated by 90 percent of the user community to be at least as easy to use as the banner system for registering for classes.
5. A database administrator shall have the ability to submit changes to a internship data set chosen from the database in a maximum of 4 minutes, with an entry change time of 2 minutes.

6. People with no training and no understanding of English shall be able to use the software.
7. A user shall be able to enter an internship feedback on the system within 15 minutes.
8. The system shall be useable by all users after 5 minutes of training.

3.4. System Help Requirements

This section describes what kind of System Help features are needed to be built into the system.

1. The software shall notify users that they must be University of Mary Washington students to input feedback into the database.
2. Users that are University of Mary Washington students, but have not had their internships evaluated as eligible by the university should be told that and given the link to the Academic Internship Contract.
3. If students run into any unknown issues there will be a contact page to report these issues.

3.5. Performance Requirements

This section describes system performance expectation levels (response times).

3.6. Scalability Requirements

This section describes how the system is expected to scale to new higher or lower levels. Both user and application scalability requirements are described here. *Data scalability is not described here as it is already described in the "data volumes" section earlier*.

3.6.0. User Scalability

3.6.1. Application Scalability

4. Interface Requirements

This section describes User and System Interface requirements for the proposed system.

4.1. User Interface Requirements

4.2. System Interface Requirements

5. Appendices

5.1 Glossary of terms related to your project

Database: information stored in a structured/organised way so it can be easily accessed, searched, managed and updated.

5.2 Author Information

Christopher Vasquez: pages 1 through 6

Daniel Schaub: pages 7 through 10

Jorge Contreras Mendez: pages 11 through 15

5.3 Additional Requirements

Figure 1: Example of how the website should look, assume near-literal

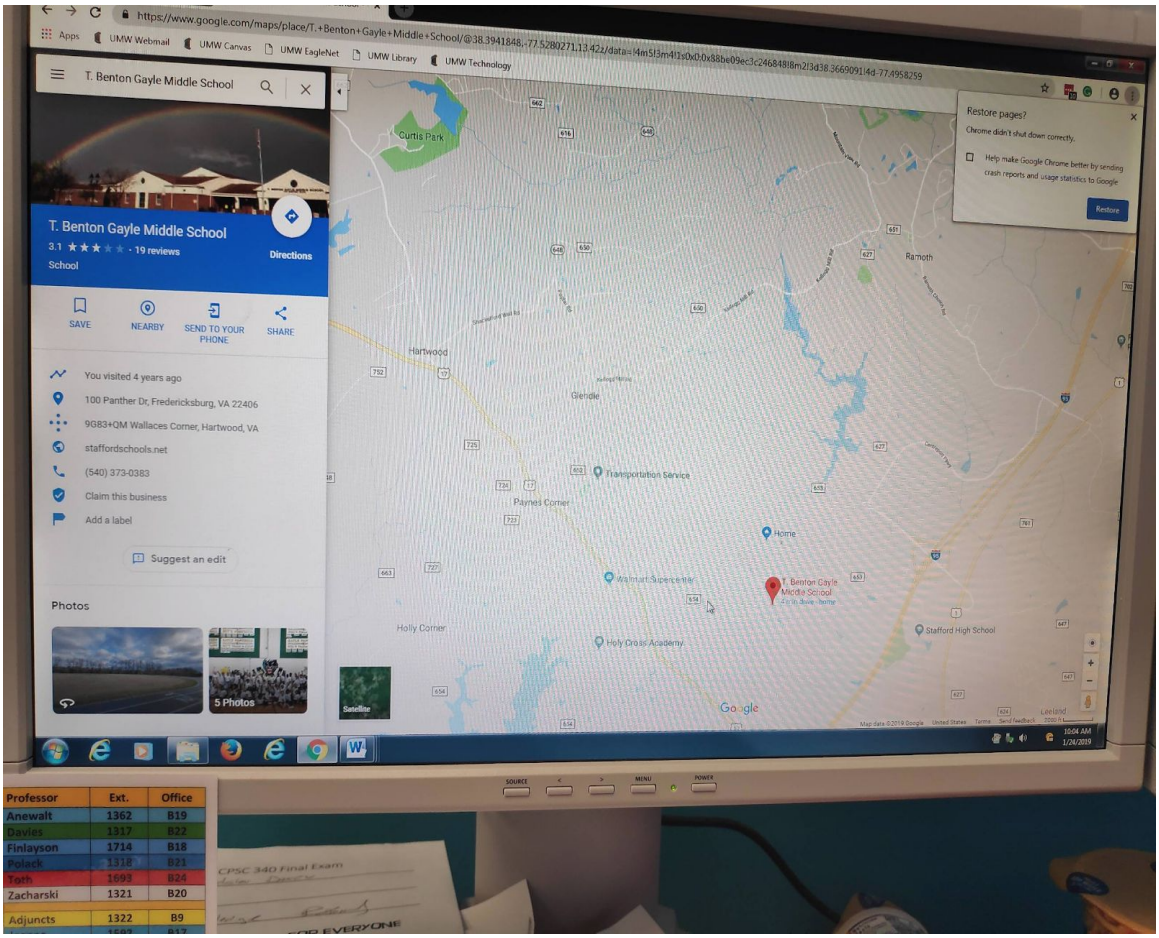


Figure 2: Basic Depiction of the layout of the website. Rough drawing, not to scale.

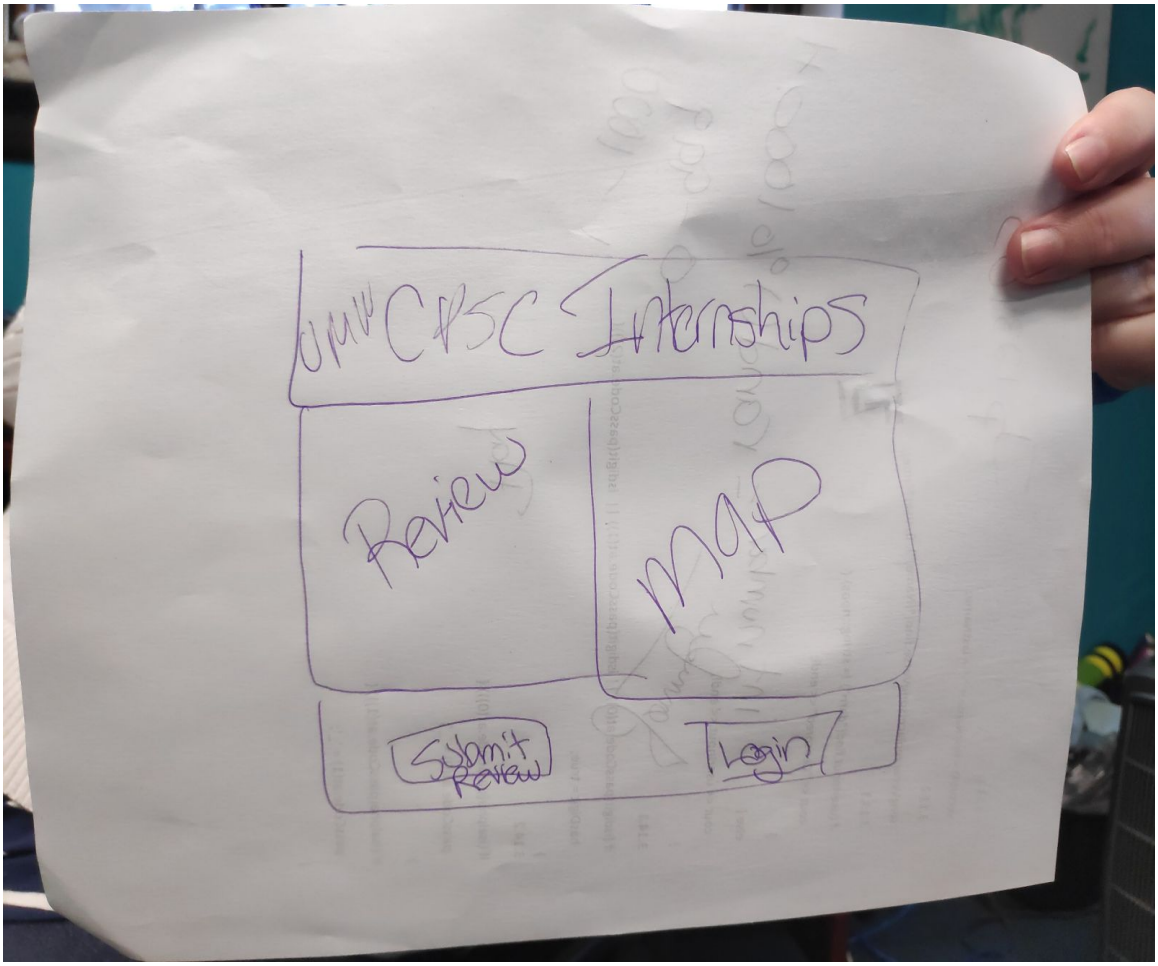


Figure 3: Added a ‘List of Internships’ to the top of the Review box. Rough drawing, not to scale.