# Test Plan

**for**

# Computer Science Internship Website

**Prepared by Clare Arrington, Harrison Crosse, John Herrin**

**University of Mary Washington**

**March 2019**

# Table of Contents

# 1. Introduction

## 1.1 Purpose

This test plan document serves as the guidelines for the testing team of the computer science internship website. This test plan was created after reviewing the plan and requirements outlined in the project plan for the internship website.

## 1.2 Scope

Our system's primary goal is to serve as an internship portal for computer science students from the University of Mary Washington. It will be a small website that allows students to post reviews about their previous internships and find information on companies where other students have had internships based on recency and location. Our client, Dr. Jennifer Polack, is a computer science professor who intends to serve as the administrator of this site.

## 1.3 Document Overview

The 'Project Description' (Section 2) gives an overview of our project by discussing the client, users, functional requirements and general constraints. The 'Test Plan' (Section 3) will provide a detailed summary of how the software will be tested to ensure proper functionality. The 'Test Procedure (Section 4) will list specific test cases to be executed. The 'Appendix' (Section 5) will catalog, define, and provide additional information for any terms contained within this document that might require further explanation.

# 2. Project Description

## 2.1 System Overview

The system is a website where UMW students are able to see and create reviews of computer science internships. Users will be able to create accounts if they own a UMW email. Subsequently, these users will be able to put pins on a virtual map at the location of the company they worked at. This will provide a base to write a review of their experiences at the internship. Only logged in users will be able to post reviews, however, any visitor to the site will be able to view the interactive map and click on pins to view the reviews for that specific company.

## 2.2 Client Characteristics

Dr. Jennifer Polack is a computer science professor who noticed the lack of easily accessible information about past internships that students have worked for. She was aware of the features she wanted from the site as well as the general design. We will be using a hosting

method she is not familiar with, so one of our challenges at the end will be in preparing her to take control of it once the project is completed.

## 2.3 User Characteristics

The intended end users of the system are UMW computer science students that wish to gain more information about internships experiences at different hiring companies. Additionally, users will be students that share information about their personal experience as interns. We are assuming these students, as computer scientists, are fairly tech-literate. Because of that, we believe they will be knowledgeable with basic web interfaces, can fill out a form, and can navigate a search setup similar to Google Maps.

## 2.4 Functional Requirements

The design allows a potential user to register with an email address ending in @mail.umw.edu or @umw.edu. The email address is verified through email. Once a user's email has been verified, the ability to login and submit reviews is granted.

Authenticated reviewers have access to fill out an internship review form. The form contains different categories for rating their internship experience. Below are the currently planned questions and their respective input type.

General Information:

| | |
|---|---|
| Company Name | written input |
| Rating | 1-5 scale |
| Location | geomarker, lat/long |
| Start / End Date | month + year) |
| Industry | checkboxes - 3 max;<br>Government Contracting, Education,<br>IT and Services, Research, Web, App,<br>Software Development, Hardware / Systems |
| Summary | written input |

Interview Process:

| | |
|---|---|
| Length | < 1 month, 1 - 3 months, 3 - 6 months, 6+ months |
| Format | Phone, Video, In-Person, Other |
| How Obtained | Referral, Online Application, Job Fair, Other |

Employment:

| | |
|---|---|
| Tools / Languages Used | input, potentially checkboxes |
| Top 3 Skills Learned | input, potentially checkboxes |
| Environment | open office, desk, lab, from home |

Employer contact information will be requested from the reviewers; information will not be publicly available without approval. Once the form is completed, an email will be sent to the employer requesting approval to display contact information on the UMW Internship Website. They will not be informed of the contents of the review. If the contact does not approve of their information being provided by the website, the review will advance with no contact information attached. Additionally, submitted review forms are placed in a pending state, waiting for administrator approval.

The administrator approval is needed to make the review information publicly available. Our client will have a management page that she can access when logged in where she can view pending reviews. She will be able to see the contents of an individual review when selected and can decide to edit the contents before publishing. If she does not wish to show a review, she can choose to deny it. Users will be notified once their review has been acted upon, regardless of decision.

As a guest user, past UMW internship experiences are searchable on the Internship Website. The guest user can search by location, tags, and/or company. The search results are displayed by most recent review first. On the right of the screen, a map is shown with dropped pins of internship job site locations corresponding to the listed results on the left side. If the guest user needs to submit a review of a past internship, the user selects the login button and uses previous credentials. If credentials have not been established, the user selects the request access button located next to the login button on the bottom right of the screen.

### 2.4.1 Internship Reviews

The internship reviews are the core data of the site. Every review available will be displayed in a list format on the main page, Figure 1, where users can scroll through to search for anything interesting. Figure 2 shows how users can search by details like company name or location to be given a more filtered group of reviews. If a user wants the full information available, they can select one review and view it such as in Figure 3.

#### 2.4.1.1 Review List

1. As a user, I want reviews sorted chronologically, so I can see the most recent activity.
2. As a user, I want to see the company name, rating, tags and location of each internship.
3. As a user, I want to search by any feature in 2.4.1.1.2 to find relevant reviews for me.
4. As a user, I want to see reviews filtered by my search query.
5. As a user, I want to see a specific list of reviews when a company is selected.

#### 2.4.1.2 Specific Review

1. As a user, I want to select one review and see programming languages used, length of internship, likes, type of environment (desk, open space, etc.), prerequisites taken before getting the internship, paid / unpaid status, valuable skills learned, and a summary.

2. As the administrator, I want most values to be from a standard selectable list, so the reviews are more uniform.

## 2.4.2 User Capabilities

This section covers the distinction between actions that the admin, guests, and logged in users are capable of doing.

#### 2.4.2.1 Guest User

1. As a guest user, I want to see reviews.
2. As a guest user, I want to search for internship reviews.
3. As a guest user, I want to log in so I can have access to the internship review form.
4. As a guest user, I want to request access so that I can join the internship review group.

#### 2.4.2.2 Logged In User

1. As a logged in user, I want to be able to submit an internship review form for approval.
2. As a logged in user, I want to log out of the system.
3. As a logged in user, I want to be reminded that reviews are approved by UMW faculty.

#### 2.4.2.3 Admin User

1. As an admin user, I want to be able to approve/edit/deny internship review posts before letting them become publicly available.
2. As an admin user, I would like to privately store the contact information of company employees if they approve.

## 2.4.3 Map Functionality

The map is meant to directly improve the search experience of users, so they can reference internship locations immediately without the use of another tool.

1. As any type of user, I want to be able to view a map with pins designating the location of companies which have reviews written about internships.
2. As a logged in user, I want to to place pins on the map where I have been an intern.
3. As any type of user, I want to click an existing pin in order to alter the reviews displayed on the side to match the company selected.
4. As any type of user, I want to zoom in or out of an area on the map in order to display the reviews relevant to the area selected.
5. As an admin user, I want to be able to remove unnecessary pins from the map.
6. As an admin user, I want to be able to add required pins to the map.

## 2.4.4 User Interface

It is expected that there will be at least three separate web pages that accomplish different goals. These can be viewed in the wireframe section 2.4.4.4.

#### 2.4.4.1 Splash page

1. As any type of user, I want to be initially shown the splash page on entering the website.
2. As any type of user, I want the splash page to display reviews, a map, and login/registration functionality.
3. As a guest user, I want to be able to create an account with my UMW email.
4. As a registered user, I want to be able to log in from the splash page.

#### 2.4.4.2 Admin page

1. As an admin user, I want to view approved and unapproved reviews on a page that only admins have access to.
2. As an admin user, I want to approve/disapprove reviews with a check box located near the review.

#### 2.4.4.3 Review Page

1. As a logged in user, I want to submit a review with a form located on the review page.
2. As a logged in user, I want to fill out the information covered in Requirements 3.1.2-3 and submit the form for admin approval.

#### 2.4.4.4 Wireframes


Figure 1: Splash Page

This is the primary page that users will be on and serves as the framework for every other page. It features a list of reviews on the left side and a map component on the right. Users can log in or register in the top right corner. They can search or, if logged in, can submit a review. The final button simply links to the computer science wiki which is discussed in 4.3.

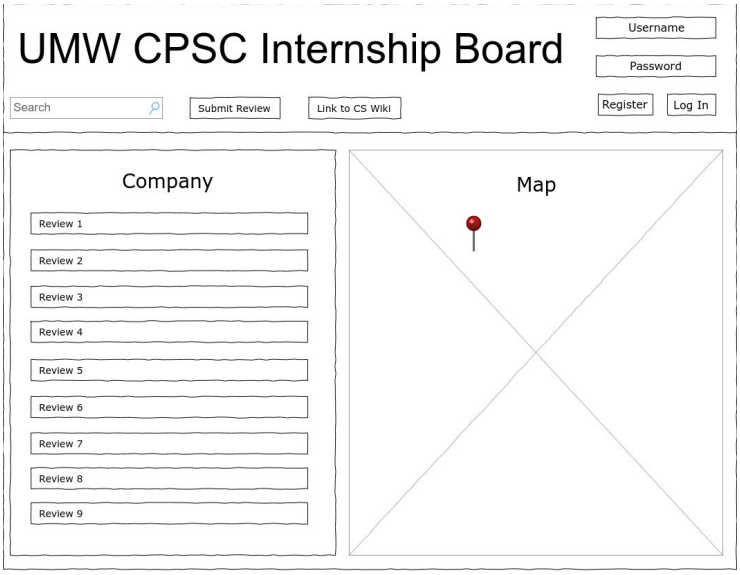


Figure 2: Selected Pin / Searched Company

This shows how the left list changes based on map input.

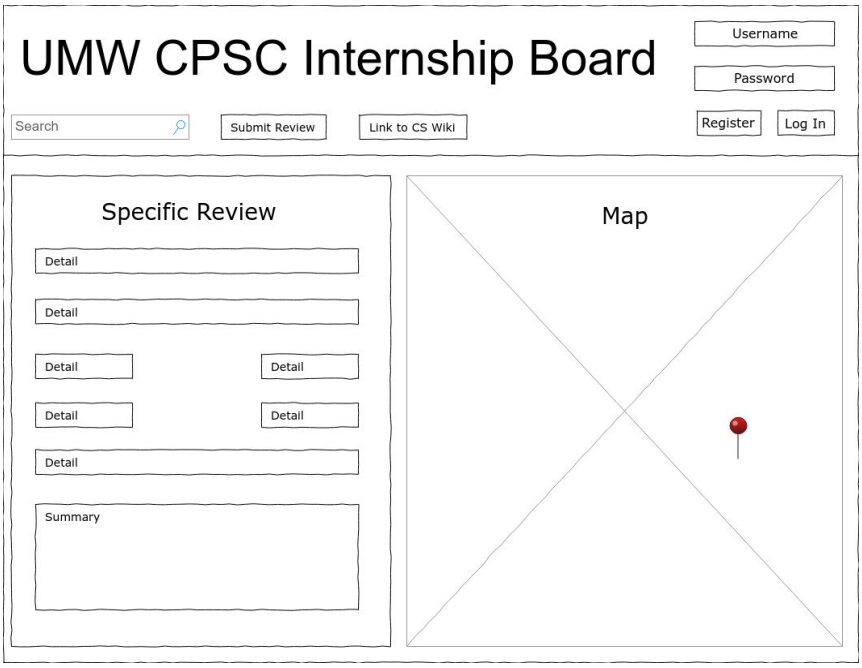


Figure 3: Selected Review

The left bar can also change to show the full information of a single review.


Figure 4: Admin Page

The admin page features a list of pending reviews and a view of a selected review that our client can edit.

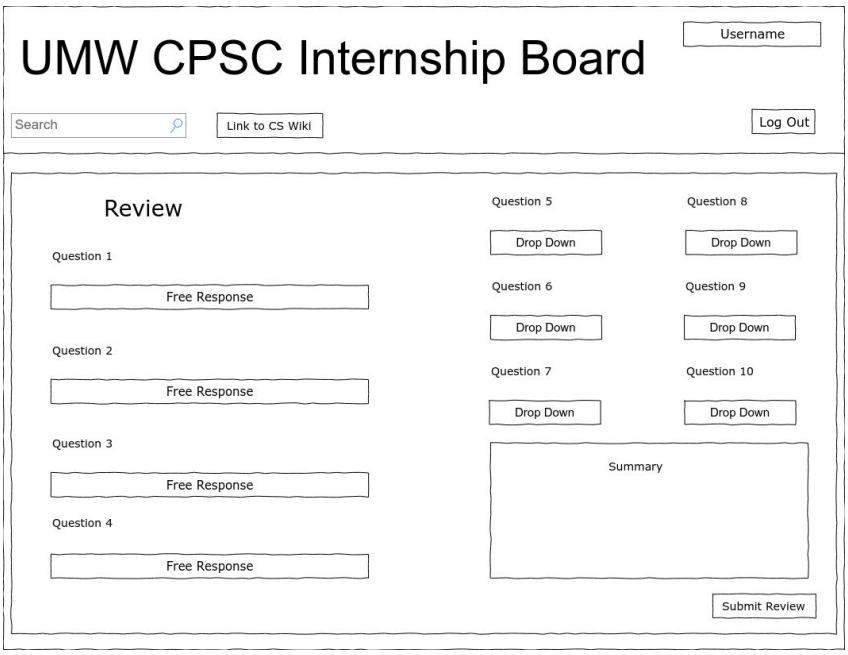


Figure 5: Review Submission Page

This page requests information from the user when submitting a review.

### 2.5 General Constraints

The project team recognizes that with the budget constraints and limited time to implement a secure website, the following assumptions are made:

- The use of a map API, database management system and web hosting needs to be between $0-20 per month total. If this project is successful, we assume we can maintain this budget past any discounted prices given during a limited trial period.
- Software security is important, however, given the time period needed to complete the project and the required personnel/skills to actively monitor and secure such a site may not be present. This project design assumes that no PII or other sensitive data will be stored on this database.

# 3. Test Plan

## 3.1 Testing Strategy

Our testing methodology has been designed with the intent of providing adequate instructions for the both the testing team and the implementation team as they progress through the testing of each feature the customer has requested. We have ordered our testing procedures (Section 4) by subsystem names which refer to each major feature the customer has requested for the final product. A brief overview of the testing approach for each subsystem are as follows:

### 3.1.1 Hosting Subsystem

- Tests contained within this subsystem category will focus on ensuring that the website is accessible to the public.

### 3.1.2 Reviews Subsystem

- Tests contained within this subsystem category will focus on tests related to the reviews that are displayed to users, administrators and the general public.

### 3.1.3 Login Subsystem

- Tests contained within this subsystem category will focus on all features related to the login process and ensuring that only approved users have access to certain pages of the domain.

### 3.1.4 Submission Subsystem

- Tests contained within this subsystem category will focus on the creation of new reviews by our users, as well as the process by which they are submitted to our database.

### 3.1.5 Admin Subsystem

- Tests contained within this subsystem category will focus on the capabilities that should be provided to admin users as defined by our client. These capabilities include features such as the ability to approve reviews for admission to the database, as well as the ability to reject reviews.

## 3.2 Testing Resources and Staffing

This section will help to explain the requirements for successfully testing the software that is under development. Here we will identify specific resources that need to be acquired and setup. This section will also address which teams will be responsible for specific tests that have been detailed in our test procedures located in Section 4 of this document.

### 3.2.1 Hardware Requirements

- There are no specific hardware requirements necessary to complete the testing procedures. However, it should be noted that the hardware you are using should be able to host an operating system that is capable of installing the software listed in section 3.2.2. It will also be necessary for all teams performing testing to have networking cards installed in their personal hardware that is capable of accessing the internet. Cell phones may also be used as hardware for testing, as the appearance of the website on mobile devices will be important in later stages of development.

### 3.2.2 Software Requirements

- It is recommended that each member of the testing team have at least two web browsers installed to test the functionality and appearance of the website on different platforms. We are suggesting that these two browsers be Chrome and Firefox as they seem to have the highest market share by users worldwide at the moment. Any testing team members that are operating on a Windows based OS should perform testing within the Internet Explorer or Microsoft Edge browsers as well.

### 3.2.3 Testing Team Responsibilities

- Due to the scheduled availability of the testing team, there will be limits to what features can assigned for testing at these specific dates. Only features that have been implemented can be tested successfully by either team at this time. To address this concern, section 4 (Testing Procedure) will have a column labeled "Status" which will indicate whether a feature is ready for testing. The testing team will be responsible for all items that are marked as "R" (for ready). Each row should provide specific details on how to accomplish testing for that task.

### 3.2.4 Integration Team Responsibilities

- Integration team members are responsible for the testing of any items in section 4 (Testing Procedure) that has a status of "NR" or "WIP". These items are not yet ready for testing and will be implemented and rigorously tested at a later date.

## 3.3 Test Work Products & Record Keeping

We will be relying on two methods for both teams to use to keep record during the testing process. Firstly we will rely on the issue tracking system that is available for GitHub projects. This allows anyone, including individuals not on the project, to report issues like bugs, desired feature enhancements, and general questions. It lets you assign specific project members to tasks, which can make it easier to keep track of incomplete requirements. Additionally, one can provide photos when reporting, so individuals can further illustrate the issue. This works well for our implementation team as a method of concrete record keeping. since we already have a channel for providing more abstract and conceptual feedback to one another.

For the testing team we will be working with, we needed a way for them to communicate with us in a similar open-ended format. Prior to viewing our test cases, we plan to have the test team members do a blind test so they are able to give their opinions as a brand new user of the system. Their exploration will be guided by a document that asks them to perform a few primary objectives such as signing up for the site, logging out and back in, and submitting a review. Then, we expect them to report back any complications and thoughts they had while using the system. For anything clearly an issue, they could choose to report these to GitHub as well.

Following the open-ended portion of the document, the individual would go through the test procedure below. For each test that is ready, we request that they report complications and opinions. The only information that definitely needs to be noted for each test is whether or not there is deviation between the expected results and the actual output, and if so, what that difference is.

## 3.4 Testing Schedule

As explained in the strategy section, we will have two test phases. The first will begin next week with the official testing team. This group of 3-4 members are anticipated to have about 3 hours over the course of 2 separate days to work through our test document. We believe between setup, blind exploration, and our currently ready cases that should cover the full time period. However, there may be some progress during the two day gap that we would like tested. If that happens, we will update the team which cases have moved from either the WIP or NR status to R status. If there is additional time available, they can complete these additional tasks.

The second phase will be conducted by the implementation team. We intend to test the cases covered by the testing team again as we continue with our development, to avoid bugs being introduced into parts of the system that are closer to being complete. There is also a plan for us to test the portions of our project that we have not gotten to or cleaned up enough for others to test yet. Since we are not working with a completed project, there are sections of the website like the full map capability and review organization that will not be ready in time for next week. Our group will have to pay more attention to these components, as we will be the only ones validating them. This testing is intended to continue regularly until the end of development.

# 4. Test Procedure

## 4.1 Alpha Testing

All of the tests listed in this section are intended to simulate every aspect of the user experience. The Test Number indicates which test is being performed, and serves for record keeping. Subsystem identifies which portion of the project is being tested, and the Purpose states why the test is being performed. The URL and Test Case Data sections specify details about where on the site the test takes place, and what data is associated with that test. Expected Results lists the expected outcome of the test if everything is performing as designed. The Status column lists the state that we expect each feature to be in during our testing week, with 'R' signifying ready and testable, 'WIP' signifying that the feature is under development, and 'NR' signifying that the feature has yet to be implemented and is unable to be tested.

| Test Number | Subsystem | Purpose | URL | Test Case Data | Expected Results | Status |
|---|---|---|---|---|---|---|
| **A-01** | Hosting | Test that website loads | http://polack-internship.appspot.com/ | N/A | Website loads | R |
| **A-02** | Reviews | Test that the review list loads | http://polack-internship.appspot.com/ | N/A | A list of reviews should appear on the left side of the website | R |

| | | | | | | |
|---|---|---|---|---|---|---|
| **A-03** | Reviews | Test that specific reviews are available by clicking on one in the list | http://polack-internship.appspot.com/ | N/A | More detailed information on a review appears when one is selected | NR |
| **A-04** | Reviews | Test that the reviews can be filtered by clicking on the tags of a specific review | http://polack-internship.appspot.com/ | N/A | The review list gets filtered by the selected tag | NR |
| **A-05** | Reviews | Test that the reviews can be filtered by one of the filter dropdowns | http://polack-internship.appspot.com/ | **Filter:** Paid | The review list gets filtered by the selected tag | NR |
| **A-06** | Reviews | Test that the reviews can be searched | http://polack-internship.appspot.com/ | **Search:** IBM | The review list shows results matching the search term | NR |
| **A-07** | Reviews | Location Filtering | http://polack-internship.appspot.com/ | **Location:** Fredericksburg, VA | The review list shows reviews within a 25 mile radius of Fredericksburg | NR |
| **A-08** | Reviews | Test that the Google Maps API is working | http://polack-internship.appspot.com/ | N/A | Map is visible on home page and displays a location near the user | R |
| **A-09** | Reviews | Test that the pins load on the map | http://polack-internship.appspot.com/ | N/A | Pins are visible on the map | WIP |
| **A-10** | Reviews | Test that the user can scroll around the map | http://polack-internship.appspot.com/ | N/A | Map scrolling works and loads new pins | R |
| **A-11** | Reviews | Test that the reviews can be filtered by clicking on a location pin | http://polack-internship.appspot.com/ | N/A | Clicking on a pin brings up all the reviews for that location | NR |
| **A-12** | Login | Test that you are unable to see the submission page without an account | http://polack-internship.appspot.com/submit | N/A | Redirect to home page | R |

| | | | | | | |
|---|---|---|---|---|---|---|
| **A-13** | Login | Test that you are unable to see the admin page without an account | http://polack-internship.appspot.com/admin | N/A | Redirect to home page | R |
| **A-14** | Login | Test that the login drawer works when the icon is clicked | http://polack-internship.appspot.com/ | N/A | Login drawer appears on the side of the page | R |
| **A-15** | Login | Test that user must have email to register | http://polack-internship.appspot.com/ | **Email:** N/A **Password:** Password1 | Invalid email message | R |
| **A-16** | Login | Test that only UMW emails are accepted when registering | http://polack-internship.appspot.com/ | **Email:** test@gmail.com **Password:** Password1 | Invalid email message | R |
| **A-17** | Login | Test that user must have password to register | http://polack-internship.appspot.com/ | **Email:** test@mail.umw.edu **Password:** N/A | Invalid password message | R |
| **A-18** | Login | Test that only strong passwords are accepted when registering. | http://polack-internship.appspot.com/ | **Email:** test@mail.umw.edu **Password:** password | Invalid password message | R |
| **A-19** | Login | Test that registration works | http://polack-internship.appspot.com/ | **Email:** your umw email address **Password:** your password | Successful registration message | R |
| **A-20** | Login | Test that you are unable to visit the submission page without a verified email | http://polack-internship.appspot.com/submit | **Email:** your umw email address **Password:** your password | Redirect to home page | R |
| **A-21** | Login | Test that logging in works | http://polack-internship.appspot.com/ | **Email:** your umw email address **Password:** your password | Submission link appears on navbar | R |
| **A-22** | Login | Test that verified users can see visit the submission page | http://polack-internship.appspot.com/submit | **Email:** your umw email address **Password:** your password | Submission page loads | R |

| | | | | | | |
|---|---|---|---|---|---|---|
| **A-23** | Login | Test that verified non-admins cannot see the admin page | http://polack-internship.appspot.com/admin | **Email:** your umw email address **Password:** your password | Redirect to home page | R |
| **A-24** | Login | Test that logging out works | http://polack-internship.appspot.com/ | N/A | Registration | R |
| **A-25** | Login | Test that the admin login works | http://polack-internship.appspot.com/ | **Email:** harrison.crosse@gmail.com **Password:** Password1 | Submission and admin links appear on navbar | R |
| **A-26** | Login | Test that the admin can visit the submission page | http://polack-internship.appspot.com/submit | **Email:** harrison.crosse@gmail.com **Password:** Password1 | Submission page loads | R |
| **A-27** | Login | Test that the admin can visit the admin page | http://polack-internship.appspot.com/admin | **Email:** harrison.crosse@gmail.com **Password:** Password1 | Admin page loads | R |
| **A-28** | Login | Test the the password reset email works by resetting it and logging in | http://polack-internship.appspot.com/ | **Email:** your umw email address **Password:** your new password | User is able to login with new password | R |
| **A-29** | Login | Test that user cannot register with an email already in use | http://polack-internship.appspot.com/ | **Email:** your umw email address **Password:** your password | Email already registered message | R |
| **A-30** | Submission | Test that each dropdown on the submission form works | http://polack-internship.appspot.com/submit | N/A | All dropdowns have items, can an individual item can be selected | R |
| **A-31** | Submission | Test that a form can be submitted with skipped optional input | http://polack-internship.appspot.com/submit | N/A | Successful form submission message | R |

| | | | | | | |
|---|---|---|---|---|---|---|
| **A-32** | Submission | Test that a form cannot be submitted with skipped required input | http://polack-internship.appspot.com/submit | N/A | Required field message | R |
| **A-33** | Submission | Test that the location field on the form does address autocompletion | http://polack-internship.appspot.com/submit | N/A | Addresses are autocompleted as typed in | WIP |
| **A-34** | Submission | Test that all open input fields don't break from miscellaneous input | http://polack-internship.appspot.com/submit | N/A | No page crashes from input on the submission | R |
| **A-35** | Admin | Test that submitted review appears on admin page for approval | http://polack-internship.appspot.com/admin | N/A | Submitted review is present on admin page | NR |
| **A-36** | Admin | Test that submitted pins don't appear on the map without approval | http://polack-internship.appspot.com | N/A | Pin from submitted review is absent from map until approved | NR |
| **A-37** | Admin | Test that the pins from submitted and approved reviews are added to the map | http://polack-internship.appspot.com | N/A | Pin from approved review is present on map | NR |
| **A-38** | Admin | Test that submitted reviews don't appear on the review list without approval | http://polack-internship.appspot.com | N/A | Submitted review is absent from review list until approved | NR |
| **A-39** | Admin | Test that a submitted and approved review appears on review list | http://polack-internship.appspot.com | N/A | Submitted review appears on review list after approval | NR |

| | | | | | | |
|---|---|---|---|---|---|---|
| **A-40** | Admin | Test that reviews can be edited | http://polack-internship.appspot.com/admin | N/A | Pending review shows changes from edit | NR |

# 5. Appendix

## 5.1 Glossary

**Administrator User** - A user with administrator privileges, typically a UMW faculty member. The administrator manages the site and approves/denies registration forms and review forms.

**API -** Application Programming Interface

**Authenticated Reviewers** - Users approved to sign into the site. These users can only submit an internship review form for administrator approval.

**CPSC** - Computer Science

**Guest User** - A user not logged into the site. Permissions - View publicly available data.

**Necessary Pins** - A pin that is required to be placed on the map in order for a review to be displayed.

**PII** - Personal Identifiable Information

**UMW** - University of Mary Washington

**Unnecessary Pin** - A pin on the map that has no review linked to it.

## 5.2 Author Information

Clare Arrington handled the cleanup of 2.4, and wrote sections 3.3 and 3.4. Harrison Crosse wrote 1.1 and 4.1 and dealt with miscellaneous reformatting. John Herrin wrote sections 1.3, 3.1, and 3.2.

## 5.3 Additional Documents

The record keeping document for the testing team can be found at this link: https://docs.google.com/document/d/1slT1xc4dji9y_lzYujoiTfpojtly4bzIw6vnpMR7yDk/edit?usp=sharing

Our client suggested that we link to the Computer Science MediaWiki, so students can find posted internships.
http://cs.umw.edu/mediawiki/index.php/Career_and_Internship_Information is the link for this specific webpage.