Product: S.S. Media

Client: Doctor Judith E. Rosenbaum

Black Bear Analytics

Abdullah Karim | Colleen DeMaris | Griffin Fluet | James West | Ryan Handlon

October 27, 2020

Revision History

Version Number	Release Date	Description
Version 1.0	10/24/2020	Original Release

System Requirements Specification

Table of Contents

1. Introduction

The purpose of this section is to introduce the reader to both this document and the system requirements for the project, detailing introductory information that will be useful for the reader to know.

1.1 Purpose of this Document

The purpose of the SRS is to clearly outline the features agreed upon by both our client (Doctor Rosenbaum) and Blackbear Analytics for the S.S. Media scraping tool. This document will cover the scope of the product and detail all requirements of the application. This includes functional and non-functional requirements, user interface design, required deliverables, and open issues.

The intended readership of this document includes Developers, Testers, the Client, and End Users. Developers will review this document to more fully understand the product, what must be developed, and help guide their efforts in its creation. Testers can use this document to help guide them in their testing of the application helping guarantee proper functionality of the product. The Client will read this document to have a greater understanding of the product they commissioned as well as guarantee that their development team understands what they need out of the product. The end user would review this document to better understand the product's functionality and intended purpose.

1.2 References

This document does not include any cited information and does not require any references.

1.3 Purpose of the Product

The client for this product is Associate Professor and Chair of the Department of Communication and Journalism Doctor Judith E. Rosenbaum at the University of Maine in Orono. A great deal of hers and her grad student's research involves the analysis of social media and how people use it. Two notable examples of this are the Boycott Nike hashtag from late 2018 or current research on the analysis of people's perceived risk when taking pictures at national parks, and whether they acknowledge it. An important aspect of this research is data collection, which requires a way to scrape and compile information based on specific hashtags, times, phrases, or location from social media sites. Doctor Rosenbaum has had tools in the past that would do this data scraping for her, however those tools were complex and eventually became outdated. She now needs a new up to date scraper that is easy for the end user to understand.

1.4 Product Scope

The scope of this project is to provide a web-based tool that can scrape information off of social media websites. The application will have a user-friendly design allowing them to login and scrape information from any supported social media platform. Scrapeable websites shall include Twitter and Instagram exclusively. Should time permit, scraping of Reddit, Facebook, Snapchat, and TikTok will also be implemented in that order. The user will be able to scrape for information based off of hashtags, dates, locations, and key words or phrases. The tool will then compile the information into an easily manageable format and provided to the end user. In the base version of the scraping tool, the application will not provide any analysis or manipulation of information after it has been gathered. As can be seen in Figure 1, the system will provide all the functionality required to manage user accounts, login, select scraping specifications, scrape data, and downloading data. Outside actors of the system will include: Researcher, Administrator, and Unauthorized User, the social media platforms being scraped which include but are not limited to Twitter and Instagram, and the storage location of scraped data.
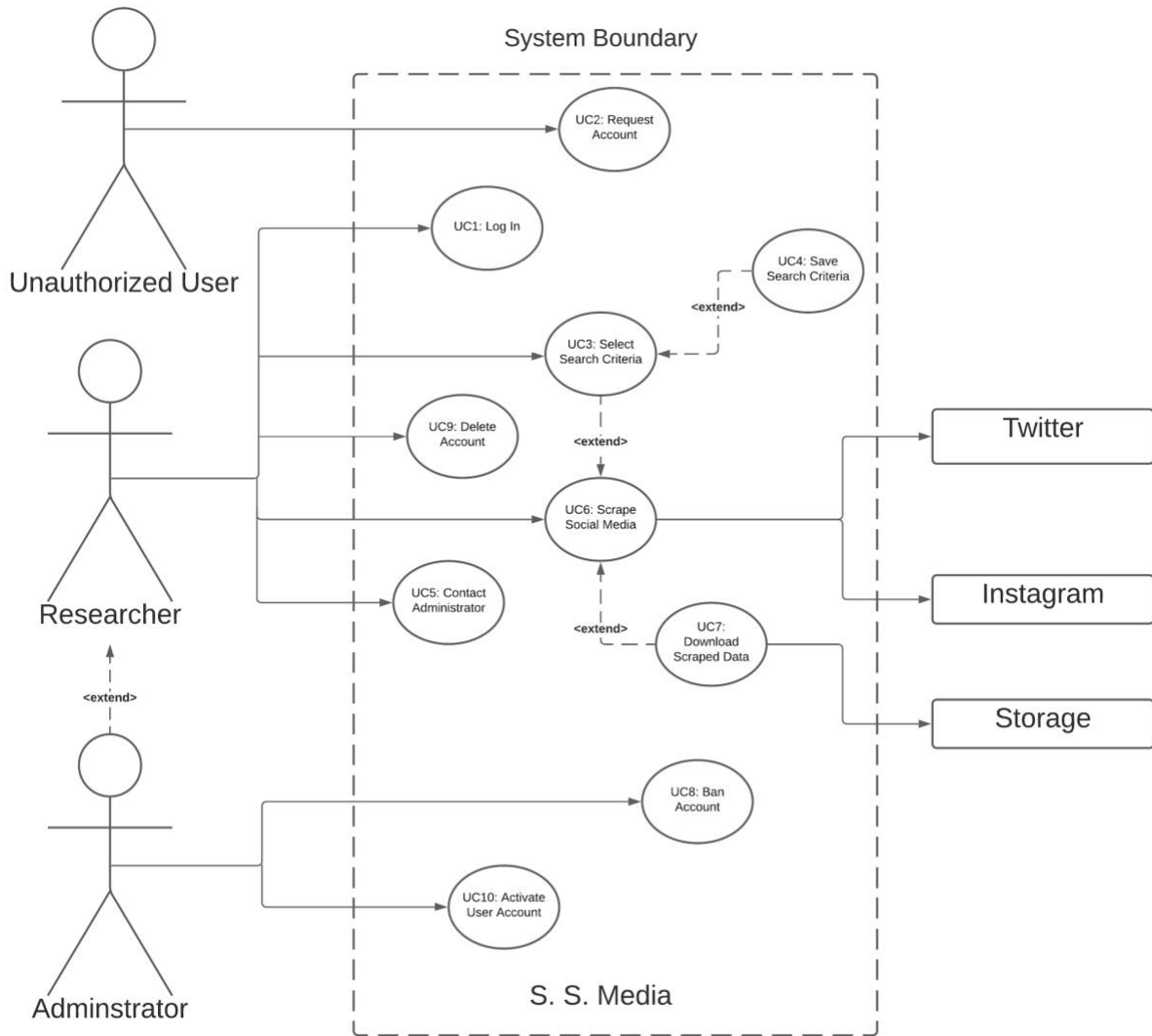
Figure 1: Use Case/Context Diagram of the S.S. Media Application

2. Functional Requirements

This section contains use cases that outline the functionality of our program. Per use case, there is a corresponding acceptance test titled in the use case description. The steps for each use case is written out at the end of the section for Functional Requirements. Functional Requirements are adapted from elicited user stories written while talking with the client about their needs. Use cases for open issues relation to functionality have no corresponding use case but will be included at the time of closing the issue. Post minimum viable product (MVP) functionality is not included in this iteration of the SRS as they're open issues that are undergoing scope definition. For acceptance tests related to each functional requirement, refer to the list of tests in section 3.2 Acceptance Tests.

2.1 Use Case Descriptions

Number	1	
Name	Log In	
Summary	This use case addresses security for our system. Users must log in with a username and password combination that's given to them by an administrator upon request and approval of an account.	
Priority	4	
Preconditions	The user must be on the login screen. They must also have valid credentials.	
Postconditions	Users will be logged into the Home Page which displays their scrape history, settings, and a button that allows for the setup of a scrape.	
Primary Actor	Researcher	
Secondary Actors	Administrator	
Trigger	User enters the login screen	
Main Scenario	**Step**	**Action**
	1	User clicks username field. System activates the name field and allows the user to type into it.
	2	User enters a username. The system saves the information the user has entered.
	3	User clicks the password field. System activates the password field and allows the user to type into it.
	4	User enters password. As the password is being typed, the system displays it as dots to the user while also saving their input for authorization.
	5	System makes the greyed-out login button active and blue so the user can log in.
	6	User clicks the login button. If the user's credentials are valid, they are logged in.
Extensions	**Step**	**Branching Action**
	6a	User failed authentication: Else, red text appears above the username field stating: "Invalid credentials. Please try again or contact the administrator."
Open Issues	Refer to Open Issue 2	

Number	2	
Name	Request Account	
Summary	Users request an account by clicking a button and filling out/sending a form for administrator approval.	
Priority	4	

Preconditions	User needs an account	
Postconditions	User submits a form for the user account that's sent to the administrator.	
Primary Actor	Researcher	
Secondary Actors	Administrator	
Trigger	Click the Request Account button from the Login window.	
Main Scenario	**Step**	**Action**
	1	User navigates to the Login window and clicks the request account.
	2	System produces a form with details of Email, First Name, Last Name, and Password text fields.
	3	User fills out the form and clicks the Request Account button.
	4	User is prompted to verify their email and the system sends the verification email with instructions.
	5	User opens the email and clicks the link to the verification page.
	6	System verifies the user.
	7	User is prompted that their account is verified and awaiting approval from an administrator. The system adds the new user account to the Administrator's Approve Accounts list.
Extensions	**Step**	**Branching Action**
	3a	The user does not enter a valid password and is not meeting the systems password criteria. The user is prompted to enter a password with at least 1 uppercase and 1 lowercase letter, 1 special character, 1 number, and be at least 8 characters long.
Open Issues	Refer to Open Issue 2	

Number	3	
Name	Select Search Criteria	
Summary	User selects their search criteria such as hashtags, locations, platform being scraped, and date range.	
Priority	5	
Preconditions	User must be logged in on the home screen.	
Postconditions	Users have a set of search criteria that they can then use to scrape data and/or save the search criteria.	
Primary Actor	Researcher	
Secondary Actors	Administrator	
Trigger	User clicks the New Search button or clicks a saved previous search.	
Main Scenario	**Step**	**Action**
	1	User clicks the New Search button.
	2	User selects the social media platform they are scraping.
	3	System loads search criteria for specified social media.
	4	User inputs basic search criteria into associated boxes. (Hashtags, Locations, Phrases)
Extensions	**Step**	**Branching Action**
	1a	User selects a Saved Search criterion from a list of saved criteria on the Home Page.
	1a1	System loads Scrap Search Criteria page and populates it with saved criteria. Skip steps 2-4.
	3a	User selects Advanced Search

	4a1	User inputs advanced search criteria (Hashtags, Locations, Phrases, Date Range, and/or logic, etc.)
Open Issues	Refer to Open Issue 1	

Number	4	
Name	Save Search Criteria	
Summary	Users can save searched criteria for future use.	
Priority	3	
Preconditions	Users are on the Social Platform Selection screen and have selected search criteria.	
Postconditions	The search criteria of a user is saved with their information.	
Primary Actor	Researcher	
Secondary Actors	Administrator	
Trigger	Save Search Criteria button from the Social Platform Selection screen.	
Main Scenario	**Step**	**Action**
	1	User clicks the Save Search Criteria button.
	2	System creates a popup to prompt for name of the search criteria.
	3	User enters a name in a popup prompt.
	4	System saves that search criteria to the scrape history associated with the user's profile in the user database.
Extensions	**Step**	**Branching Action - N/A**
Open Issues	N/A	

Number	5	
Name	Contact Administrator	
Summary	Users having trouble logging in or who have general issues with the software can click the Contact Administrator button on the login window and submit a form to administrators so issues can be resolved.	
Priority	1	
Preconditions	User is on the Login page.	
Postconditions	User submits a form with the submit button, sending their contact information and a description of the issue to admins.	
Primary Actor	Researcher	
Secondary Actors	Administrator	
Trigger	Click the Contact Administrator Button.	
Main Scenario	**Step**	**Action**
	1	User clicks the Contact Administrator Button.
	2	System provides a form where the user can enter their contact information and description of their issue.
	3	User enters contact information and description of their issues.
	4	User clicks the submit button.
	5	System sends this information to Administrator accounts.
Extensions	**Step**	**Branching Action**
	1a	User clicks the cancel button to retract their contract form.
Open Issues	N/A	

Number	6

Name	Scrape Social Media	
Summary	The system will initiate a search for all relevant information from the selected social media platform.	
Priority	5	
Preconditions	User is on the Social Platform Selection screen and has filled in all relevant search criteria.	
Postconditions	Data relating to the search criteria will be gathered from relevant social media sites.	
Primary Actor	Researcher	
Secondary Actors	Administrator	
Trigger	Click the Perform Scrape button	
Main Scenario	**Step**	**Action**
	1	User clicks the Start Scrape button.
	2	System accesses selected social media platform.
	3	System runs algorithm scraping information with an input of selected criteria.
	4	System finishes running algorithm and prompts user to download scraped data. (see Download Scraped Data Use case).
Extensions	**Step**	**Branching Action**
	4a	User clicks the Halt Scrape button.
	4a1	System stops the scraping algorithm and returns the user to the home screen, throwing out any captured data in the process.
Open Issues	Refer to Open Issue 3	

Number	7	
Name	Download Scraped Data	
Summary	Once a scrape is finished, the data is downloaded to the user's local machine, so they are accessible to the user.	
Priority	4	
Preconditions	User has hit the Scrape button and a scrape has been performed.	
Postconditions	All information associated with a scrape is stored out to excel files and folders on the user's desktop.	
Primary Actor	Researcher	
Secondary Actors	Administrator	
Trigger	Scrape has completed.	
Main Scenario	**Step**	**Action**
	1	The system has met the search criteria's time constraints, and finishes searching.
	2	The system prompts the user if they want to download or discard the scraped data.
	3	User selects download
	4	The system downloads the data to the user's device and saves it to the Downloads folder.
Extensions	**Step**	**Branching Action**
	3a	User selects Discard.
	3a1	The system deletes the scraped data and returns the user to the home page. Skip step 4
Open Issues	Refer to Open Issue 3	

Number	8
Name	Ban Account
Summary	Administrators are allowed to ban user accounts that exist.
Priority	1
Preconditions	User must be an admin logged in.
Postconditions	A user account is added to a blacklist.
Primary Actor	Administrator
Secondary Actors	N/A
Trigger	User selects an account from the Edit User Accounts list and clicks the Ban button.

Main Scenario	**Step**	**Action**
	1	User clicks a checkbox next to a user's name.
	2	User clicks the Ban button.
	3	The system displays a verification popup.
	4	User clicks the Yes, I Am Sure button.
	5	The system removes the account from the user database.
Extensions	**Step**	**Branching Action**
	1a	User clicks the Select All checkbox.
	1a1	User clicks the Ban button.
	1a2	The System returns to the Edit User Accounts page.
Open Issues	N/A	

Number	9
Name	Delete Account
Summary	Researchers are allowed to delete user accounts that exist.
Priority	1
Preconditions	User must be a logged in researcher. User must be on the settings page.
Postconditions	A user account is deleted.
Primary Actor	Researcher
Secondary Actors	N/A
Trigger	User clicks the delete account button.

Main Scenario	**Step**	**Action**
	1	User navigates to the settings page.
	2	User clicks the Delete Account button.
	3	The system displays a verification popup.
	4	User clicks the Yes, I Am Sure button.
	5	The system removes the account from the user database.
Extensions	**Step**	**Branching Action**
	4a	User selects the "No, I am not sure" button.
	4a1	The system returns the settings page.
Open Issues	N/A	

Number	10
Name	Activate User Account
Summary	Administrators must approve user accounts when a form request is received. Approving a user account gives them access to the website beyond the login page.
Priority	2
Preconditions	Must be logged in as an Administrator and on the Edit User Accounts screen.

Postconditions	A user's verified account is activated, giving them access to the tools and functionality.	
Primary Actor	Administrator	
Secondary Actors	Unauthorized User	
Trigger	Account request form is submitted from an Unauthorized User.	
Main Scenario	**Step**	**Action**
	1	Administrator clicks the checkbox next to the new user.
		System displays information about the new user.
	2	Administrator clicks the Approve button.
	3	New user removed from the Approve Accounts list and gains researcher permissions.
Extensions	**Step**	**Branching Action**
	2a	Administrator clicks Deny button.
	2a1	New User is informed they have been denied via email, account is deleted, and removed from the Approve Accounts list.
Open Issues	N/A	

3. Non-Functional Requirements

This section of the document will detail all Non-Functional requirements for this project. All Non-Functional Requirements will be accompanied by a unique id number, a priority, a clear description, and a test that will be used to verify the requirement has been met.

3.1 Non-Functional Requirements

ID	Description	Priority	Tests
1	The system shall authenticate (or prevent authentication) to users within 5 seconds of the user clicking the login button.	5	Authorization Test
2	The system shall save the search criteria successfully 99% of the time when the user clicks the "Save Search Criteria" button.	4	Search Feature Test Scraping Test
3	The system shall return scraped, text-based information in a CSV file 99% of the time when the user clicks the start scrape button.	2	Scraping Test
4	The system shall return scraped, visual-based information 99% of the time when the user clicks the start scrape button	2	Search Feature Test Scraping Test
5	The system shall return scraped information within an overall time period of no more than 1 seconds per post being scraped	3	Scraping Test

6	The system shall load the saved search criteria successfully 99% of the time	5	Search Feature Test Scraping Test
7	The system shall provide accurately scraped data with 98% confidence.	5	Data Accuracy Test
8	The system shall be Open Source.	1	Licensing Test
9	The system shall have 99% of the information in its user database encrypted.	4	Encryption Tests
10	The system shall be compatible with popular platforms including Google Chrome, Firefox, Safari, and Microsoft Edge.	2	Compatibility Test

3.2 Acceptance Tests

Authorization Test
Corresponding Non-Functional Requirement ID's: 1
Corresponding Functional Requirement ID's:

Enter username and password. Start a timer and then click the Login button. When the user is logged in, stop the timer. If the timer is greater than 5 seconds the test fails, otherwise it passes.

Search Feature Test
Corresponding Non-Functional Requirement ID's: 2, 4, 6
Corresponding Functional Requirement Use Cases: 3

Go to a "@socialmediaplatform". Post a "@testdata" on the developer account being used with a keyword to test such as "@parameter" with a unique id that goes with the post so that it's easily identifiable post-scrape. Navigate to the social platform selection screen on the web scraper. Enter selection criteria such as "@parameter". Run the scrape and see if the information provided contains the unique id specified. Repeat per @socialmediaplatform, @testdata, and @parameter. If it succeeds 99% of the time, the test passes. Else, test fails.

Scraping Test
Corresponding Non-Functional Requirement ID's: 2, 3, 4, 5, 6
Corresponding Functional Requirement Use Cases: 6, 7

Navigate to the Social Platform Selection screen on the web scraper. Enter selection criteria. Start a timer and hit the scrape button simultaneously. Keep track of the number of requests. Once the process is done, stop the timer and check for any data produced in the downloads folder. If there is data and the total time taken for the scrape divided by the number of scrapes is under 1 second, the test succeeded. Else, test failed.

Data Accuracy Test
Corresponding Non-Functional Requirement ID's: 7

Scrape information from a social media platform. If there was an image, access the image and see if it's good quality. If it is, use the image's filename as a unique id to check the excel sheet for any corresponding text information. If there is, verify that information and the image correspond to the post on the social media platform that it's describing. If the information is just text (such as a tweet), go to that text's corresponding post on the social media platform and verify that the information is the same. If it succeeds 99% of the time, the test passes. Else, test fails.

Licensing Test
Corresponding Non-Functional Requirement ID's: 8

Open the software licensing agreement. Read it. If the license states the software is Open Source the test passes, otherwise it fails.

Compatibility Test
Corresponding Non-Functional Requirement ID's: 10

Open the web scraper in "@web browser". Run all relevant tests to the application including the Authorization Test, the Search Feature Test, the Scraping Test, the Data Integrity Test, and the testing of basic website functionality. If all tests pass, the compatibility test passes for "@web browser". Repeat test for all applicable browsers.

Encryption Test
Corresponding Non-Functional Requirement ID's: 9
Corresponding Functional Requirement Use Cases: 2, 10

Open the application and create a new account with test user information. On the administrator account, approve the account request. Log in as the new user. Request test user information from the user database. Compare the raw data received to the actual user data. If they do not match, the received data looks encrypted, and isn't in plain text, the test passes. Otherwise the test fails.

Login Test
Corresponding Functional Requirement Use Cases: 1

Get an approved account from through the administrative process. Attempt to log into the platform with fake login credentials. The system will give you an error and tell you to try again. Enter the correct credentials for a user account. The system brings you to the Homepage of the scraper tool within 5 seconds of clicking the Login button. If the system did not give an error or the correct credentials failed to log the user in, this test fails. Else, this test passes.

<u>Save Search Test</u>
Corresponding Functional Requirement Use Cases: 4

Log into the scraper platform with valid credentials. Select the New Search Button. Enter search criteria and a platform for scraping. Click the Save Search Criteria button. Fill out the form that asks for a name of the search and hit submit. Navigate to the Home Page where saved search criteria names are shown and alongside the New Search button. If the search criteria name that was saved is there, click it (Else, the test fails). If the Social Platform Selection screen is brought up and the saved search criteria are auto filled, this test passes. Else, test fails.

<u>Administration Test</u>
Corresponding Functional Requirement Use Cases: 5, 8, 9

Open the application and create a new account with test user information. Log in to the administrator account and approve the new test account request. Log in as the new test user and fill out the Contact Administrator form with sample test information. Go back to the administrator account and check for a notification with the corresponding test data. If it's not there, the test fails, otherwise continue. Ban the new test account. Try to log in to the test account. If the login is successful, the test fails, otherwise continue. On the administrator account, unban the test account. Try to log in to the test account. If login fails, the test fails otherwise continue. On the test account, navigate to the settings page. Click the Delete Account button and select yes on the Are you sure prompt. After being redirected to the login page, try to login to the test account. If the login is successful, the test fails. Else, this test passes.

4. User Interface

See "User Interface Design Document" for *S.S. Media*. This document is a deliverable due on November 22nd, 2020 and will be prototyped, with a final version not available until then.

5. Deliverables

This section of the document provides a list of all deliverable items throughout the course of this project. Each deliverable will be accompanied by information about its due date, what format it is to be delivered in, method of delivery, and any notes.

5.1 Deliverable Schedule

Name of Deliverable	Format of Deliverable, Method of Delivery	Date of Delivery to Client	Notes
Budget Estimate	.xlsx (Excel), through email and GitHub	October 14th, 2020	
Systems Requirement Specification	PDF, through email and GitHub	October 23rd, 2020	At this time Doctor Rosenbaum can look over the document and suggest any changes before signing it.
System Design Document	PDF, through email and GitHub	November 9th, 2020	
User Interface Design Document	PDF, through email and GitHub	November 22nd, 2020	
User Manual	PDF, through email and GitHub	April/May, 2021	
Administrator Manual	PDF, through email and GitHub	April/May, 2021	
Update Manual	PDF, through email and GitHub	April/May, 2021	This manual is used to find places in the code (in addition to code comments) that need to be updated if social media platforms update their APIs so that future developers can still use this program once platforms have changed. It will also outline how to use the program for new users as well as have a changelog of information
Executable Program	.src files, website link through email and GitHub	April/May, 2021	

6. Open Issues

The Open Issues section is a list of all currently known problems or questions pertaining to software requirements that need answering. Each open issue will be accompanied by a brief explanation of the issue, a timeline for when the issue will be resolved, and a specification of who is in charge of solving it.

Open Issue #1: Search Criteria
It is unknown whether the specified search criteria (Hashtags, Locations, Phrases) is viable for all social media platforms being scraped. For example, Twitter uses geolocation tags while Instagram does not all the time. There may also be special search criteria that are specific to individual social media. These search criteria need to be researched and a client meeting about the topics needs to happen. This issue shall be resolved within 20 days of the signing of this document and is assigned to Developer Griffin Fluet.

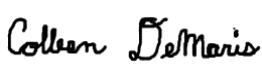Open Issue #2: Number of Users
The number of supported users at any given time is yet to be determined. Currently, we go with one user who carries the role of Administrator and Researcher, but we'd like to see if this can be expanded to support more than one user at a time. This criterion needs to be researched and a client meeting needs to happen to resolve the situation. This issue shall be resolved within 14 days of the signing of this document and is assigned to Developer Colleen DeMaris.
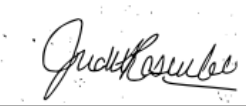
Open Issue #3: Size of Files
How much information that can be collected in a single scraping session and the potential size of that file are potential problems. Allowing for scrape sessions and file sizes that are too big could result in a download or transfer that could break something. This criterion needs to be researched and consulted on with the client. This issue shall be resolved within 14 days of signing this document and is assigned to Developer James West.

Appendix A – Agreement Between Customer and Contractor

The client (Doctor Rosenbaum) and Blackbear Analytics agree that all requirements in this document will be implemented in good faith. All requirements will be implemented as close to the specifications as possible, with margins for error being slim or none. In the event that implementation must diverge from the specifications, the development team will notify the client as soon as possible and work together with the client until a suitable agreement is reached. If the client would like to add requirements after signing this document, Blackbear Analytics bears no responsibility for the failure to implement new requirements but will attempt to add them in good faith. Should any new requirements be added, the deliverables schedule in section 5 of this document will be updated immediately to reflect changes, and the requirements discussed will be reflected in this document by the next deliverable deadline and changes will be finalized with the client.

Name	**Signature**	**Date**

Customer:

Judith E. Rosenbaum   10/25/2020

Comments:

Team:

Ryan M. Handlon   October 24th, 2020

Abdullah I. Karim   October 24th, 2020

Griffin L. Fluet   October 24th, 2020

Colleen DeMaris   October 24th, 2020

James West   October 24th, 2020

Appendix B – Team Review Sign-off

By signing your name below, you acknowledge that you are a member of Blackbear Analytics and have read the document with a solid comprehension of the scribed materials. You agree to complete all requirements stated on this document as is, in good faith. Should any new requirements come up, you agree to assist in re-drafting this document for approval as described in Appendix A. You agree that it is not required to complete requirements added after this version is released, but understand that all requirements, new or old, must be implemented in good faith.

Name	**Signature**	**Date**
Ryan M. Handlon		October 24th, 2020
Comments:		
Abdullah I. Karim		October 24th, 2020
Comments:		
Griffin L. Fluet		October 24th, 2020
Comments:		
Colleen DeMaris		October 24th, 2020
Comments:		
James West		October 24th, 2020
Comments:		

Appendix C – Document Contributions

It's worth mentioning that while each section varied in size, each team member did preliminary research that went into the decision-making process for much of this document which is a major factor considered when creating Appendix C.

Each member contributed to drafting this document evenly. Ryan Handlon created the Introduction (Section 1) and did much of the review for the document, earning a 25% share. Abdullah Karim contributed to the Functional Requirements (Section 2) and the review of the document with a percentage contribution of 25%. Colleen DeMaris wrote the Deliverables (Section 5) section of this document with a percentage contribution of 20%. Griffin created the Non-Functional Requirements (Section 3) section and had a 15% contribution to the document. James wrote out Section 4, Open Issues, Appendix A, and Appendix B which led to a 15% contribution. Blackbear Analytics as a team contributed to Appendix C to ensure transparency and talk through team dynamics.