

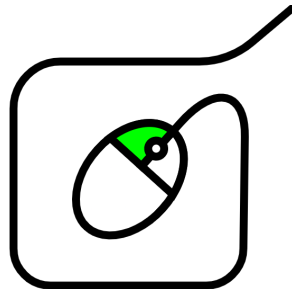
Final Report

Version 1.0

5 May 2022

SuperGeeks

CS 486C
Capstone Experience



Team Members: Tim Giroux, Kyle Austria, Gustavo Valencia, Pengfei Liu

Clients: Dr. Marco Gerosa, Dr. Igor Steinmacher

Mentor: Felicity Escarzaga

Table of Contents

1. Introduction	2
2. Process Overview	4
3. Requirements	6
4. Architecture and Implementation	8
4.1 Implementation	8
4.2 Architecture	9
5. Testing	11
5.1 Unit Testing	11
5.2 Integration Testing	11
5.3 Usability Testing	12
5.3.1 Usability Test Results #1	12
5.3.2 Usability Test Results #2	17
5.3.3 Usability Test Results #3	22
5.3.4 Usability Test Results #4	26
5.3.5 Usability Test Results #5	31
6. Project Timeline	37
7. Future Work	38
8. Conclusion	39
Appendix A: Development Environment and Toolchain	40
Hardware	40
Toolchain	40
Setup	40
Prerequisites	40
Installing	41
Testing	41
Deployment	41
Deploy Locally	41
Deploy to a Server	41

1. Introduction

Software engineering research is critical to the success of the many industries that rely on quality software products. It provides insights that are highly valuable to users and developers alike. Empirical data about what does and does not work in software engineering is very important in the world today, especially for critical software systems in the medical and structural engineering industries. Research in this field often depends on human participants. Researchers need to coordinate with software professionals in order to track the efficacy of various engineering strategies. Without this coordination, much of this critical research would be too challenging to conduct.

While researcher-participant coordination does take place in software engineering research today, it is difficult to facilitate and limited in scope. Researchers face a number of core problematic business functions:

- *Limited scale:*
Large-scale research projects could result in important discoveries but are often discarded because there is so much difficulty in obtaining participants.
- *Inefficient recruitment:*
Recruitment efforts are ineffective and create a large amount of overhead for researchers.
- *Sampling bias:*
Typically, only a subset of desired participants for a study actually respond to recruitment efforts. This introduces bias because the other desired participants will not be included in the final results of the study.

The current solutions to these problems take much more time and effort than should be necessary. Researchers might send out mass emails to software professionals based on their Github profiles, but this kind of email is often considered as spam by developers. Developers can sometimes be irritated by an email like this, or they may simply ignore it. Of those who respond, many do so only to politely decline.

Dr. Gerosa and Dr. Steinmacher, sponsors of the GeekSurvey capstone project, are researchers in the SICCS Software Engineering Research Laboratory at Northern Arizona University. They produce a large volume of software engineering research each year. This research supports the use of innovative tools, processes, and strategies to increase the quality of software products. The clients' primary means of obtaining participants for their research is mass email efforts. Email is most commonly used because research often depends on a qualified pool of subjects with very specific desired characteristics. The GeekSurvey sponsors report that this is a prominent bottleneck in their research process. They have also reported that the same problem impacts many of their colleagues. After trying existing products like Prolific and Amazon Mechanical Turk, they found that these products lack the ability to easily filter

participants based on their professional qualifications. The clients need a product that will appeal to the needs of both software professionals and researchers.

GeekSurvey serves as that product. It is a public web application in which users are able to sign up to create an account and create, manage, or participate in research studies. GeekSurvey offers the following core features to facilitate researcher and participant coordination such as:

- Two account views: researchers and participants. By default, all accounts can access researcher and participant features. These two account views are an abstraction to clarify the purpose of different GeekSurvey features. Accounts can be manually given administrative access.
- The participant view includes information about the user such as their occupation, level of education, experience with certain technologies, account balance, and a link to their Github account.
- Researchers have the ability to create studies and specify necessary criteria for participation.
- Participants are given access to studies in which they are eligible to participate in. They can receive compensation from researchers for completing the study.

GeekSurvey is a high quality, extensible web application with all of the core features needed for conducting research studies. Using a GeekSurvey study invitation link, the project clients and other researchers will be able to create, manage, and actually conduct research studies. Upon clicking the link, participants will see a landing page for the study and will be required to make a GeekSurvey account to participate. After making an account, they will be able to discover other active studies that they are eligible for on the GeekSurvey platform. Overall, GeekSurvey will improve the clients' current business process with the core features and functionalities.

2. Process Overview

The team's process involved setting up numerous standards for the team. During the Fall 2021 semester, the team met on a weekly basis for two hours, and would meet more frequently if necessary. For the Spring 2022 semester, the team met three times a week, for one or two hours depending on the assignments and tasks the team were doing. Outside of the regularly scheduled meetings, work would continue to be done asynchronously on assignments and tasks when needed.

During team meetings, the team would come together to work on a task report that outlined tasks that each team member was responsible for completing. These tasks were primarily independent work, however some tasks could require that team members work in pairs or as a complete team. Typically, the events that happened in meetings would be documented in meeting minutes documents. For the decision making process during meetings, the team would rely on a majority vote, which in our case was a 3/4 majority. In the event of a deadlock, the team leader was in charge of breaking the tie. During team meetings and other team functions, it was expected that all team members be honest and transparent. Should any disputes arise, all team members would be required to work out a solution. If the disputes could not be resolved as a team, they could be escalated to the Computer Science Capstone Organizer or other Capstone faculty.

A variety of tools and document standards were developed for the team's process. For version control, the team utilized git. The team established standards for commits, branching, and forking. An example of a commit standard was that commits should be frequent and atomic, meaning that all files related to a task in a single operation should be committed regularly. An example of a branching standard was that the main branch should contain only major changes or completed features from other branches. An example of a forking standard was that forks should be used to facilitate parallel development.

For communication, the team mainly used Discord, as it allowed the team to create a server wherein several text channels were created to keep communications organized. Discord also facilitated remote pair programming or collaboration through screen share and voice channels. If a team member could not be reached via Discord, the team utilized other means of communication, such as SMS messages and phone calls. Additionally, Discord was also utilized for issue tracking. This was accomplished by having a dedicated text channel for reporting tasks assigned and their progress status.

For word processing and presentations, the team utilized Google Drive and other Google services to organize and collaborate on documents. The team also set composition and review standards for word processing and presentations. This involved working both synchronously and asynchronously to complete assigned sections of documents and presentations ahead of the deadline. Once each team member completed their respective sections, the team would review the document/presentation in its entirety, with the intention of refining each document. A

team member would also cross-check the document against the assignment specifications or rubric to ensure that all criteria were met.

Finally, the team established roles and responsibilities for each team member. The four specialized roles include team leader, product manager, editor, and scholar. The team leader was in charge of coordinating task assignments and ensuring work progressed. They were also in charge of running meetings and making initial efforts to resolve conflicts. The product manager was in charge of managing MVP details and implementation with a focus on client satisfaction. The editor was in charge of maintaining a high standard of writing for all group documents. Finally, the scholar was in charge of keeping track of all assignment deadlines and understanding all detailed requirements and rubrics. Outside of these specialized roles, all team members were expected to contribute to software development.

3. Requirements

First and foremost, the project proposal document was a great starting point for determining which features were required for this project. Outside of the project proposal document, requirements were gathered through regular communication with the clients. The team typically emailed the clients once a week with updates on any progress made. It was through these emails that the team was able to receive clarification on any questions or concerns, as well as quality feedback on work the team was doing. Additionally, the team also met with the clients virtually on a bi-weekly basis or as needed. It was through these virtual meetings that GeekSurvey's progress was reviewed and any technical or design questions that were left after our email communication were discussed.

The requirements acquisition process ultimately resulted in three different types of requirements: functional, non-functional, and environmental requirements. Regarding functional requirements, GeekSurvey needed to support user accounts, profiles with enrollment criteria, study creation with necessary criteria for participation, study enrollment and completion for eligible accounts, and a payment system to incentivize study participation.

GeekSurvey supports generic user accounts that can be created by signing up with email or signing in with social accounts like GitHub or Google. Each account contains two different views: a researcher view and participant view. Additionally, GeekSurvey supports the creation of administrator accounts that can manipulate all studies and user profiles on the platform. Accounts on GeekSurvey are associated with a customizable profile. GeekSurvey provides a user-friendly interface for creating and customizing these accounts. The fields in the profile are used for study filtering.

Studies are the primary means of cooperation between researchers and participants on the platform. A study in GeekSurvey is a bundle of data containing everything necessary to conduct a minimum viable research study. Participants are able to use a user interface to enroll in a study. Once enrolled, they are added to a list of enrolled participants to the study. After enrolling, a user can access the study survey link. Upon completing the survey, they will return back to GeekSurvey and their participation will be recorded in the GeekSurvey backend. GeekSurvey includes a payment system to create incentives for participants to use the platform. The payment system was developed in a sandbox environment using fake external payment accounts.

Regarding non-functional requirements, GeekSurvey must respect user privacy, be accessible, and be simple to use. For privacy, GeekSurvey users only have access to information that is required to conduct or participate in research. For accessibility, GeekSurvey is able to handle simultaneous user sessions, has a limited backend delay, and works on modern browsers. For usability, GeekSurvey has a mobile-friendly frontend that allows users to do anything they are able to do on a desktop browser in a mobile browser.

Finally, regarding environmental requirements, GeekSurvey has a number of dependencies, standards it must comply with, and documentation it must come with. For dependencies, GeekSurvey is dependent on multiple Python packages and libraries that must be installed using Pip. In addition, GeekSurvey is dependent on an external survey platform for use with study conduction and participation. For standards, GeekSurvey must comply with modern web standards, such as HTTPS. Finally, GeekSurvey must be maintainable and extensible, which means that it must come with extensive documentation.

4. Architecture and Implementation

As a standard Django web application, GeekSurvey utilizes a Model View Template architecture. The Model View Template architecture provides for easy and seamless development of different view components with little to no duplication of code. GeekSurvey also utilizes Bootstrap 5 on the frontend to provide a clean visual appearance to the website. The implementation is centered around many custom user forms for editing user profiles and study fields. Users need to be able to easily edit these database items in order for us to enforce enrollment criteria on the backend. Using Django and a Model View Template architecture, features for GeekSurvey were easily implemented and developed in an organized way.

4.1 Implementation

GeekSurvey is implemented as a standard Web 2.0 application using a Django backend. Django has built in functionality for handling user accounts and login sessions. This is critical because user accounts are the core feature from which everything else is built.

Every Account has an associated Profile which can be edited by the user. This Profile contains information relating to enrollment criteria such as age, years of software engineering experience, and level of education. When a user creates a Study, they can define criteria for enrollment. GeekSurvey users whose Profiles do not meet the criteria will not be able to enroll in the Study. This is checked programmatically when serving a Study enrollment POST request, because the backend has access to the database holding all information about the Study and Profile in question.

For the purposes of discussing GeekSurvey's Study functionality, the term *Participant* will refer to a user who completes a study, and the term *Researcher* will refer to a user who creates a study.

Study creation is a simple web form. The key feature will be a link to a Google Forms survey. A major challenge in GeekSurvey's required functionality is making sure GeekSurvey tracks participation data correctly for external surveys. Researchers must include a secret code at the end of their external survey, and Participants must enter this secret code to GeekSurvey upon completing the external survey. Studies will be stored in the database with fields such as owner (Account), creation date, expiry date, enrolled participants (list of Accounts), title, description, completion code, and external survey link.

To implement a payment system (using fake/mock payments), GeekSurvey stores credentials to an external GeekSurvey payment account. User accounts have a balance starting at zero. Users can add to their balance by sending a payment to GeekSurvey. Authenticated users will be able to cash out their balance, triggering a payment from GeekSurvey to themselves. Studies will have an associated balance which will be automatically distributed to enrolled users who complete the associated survey. Study owners can add to the balance of a study by transferring their account balance into the Study balance. Study owners can also

define the payout that will be given to participants.

The user interface will be simple HTML and CSS. HTML files will be generated from templates using Django's default integration with the Jinja2 templating engine.

4.2 Architecture

GeekSurvey uses the Python Django web framework for a web application backend. Its architecture is generally the same as that of any other Django application.

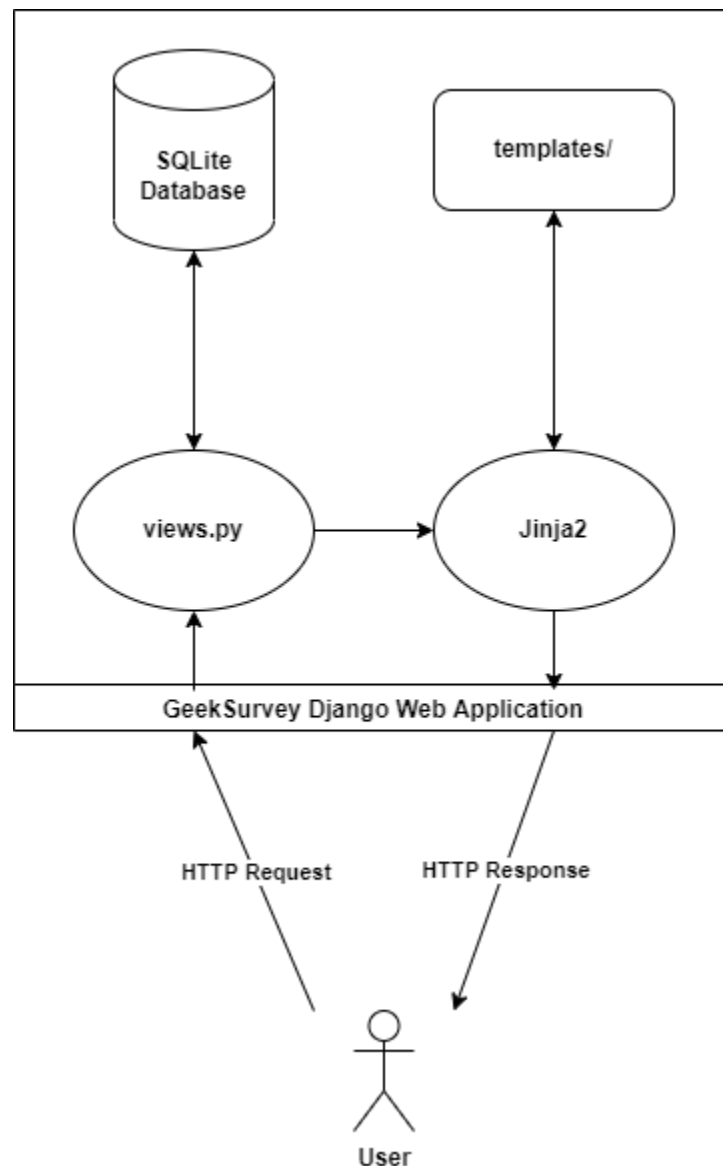


Figure 4.2.1 - Architecture Overview

GeekSurvey handles requests through custom functions in `views.py`. Upon a POST request, some data may be written to the database. Upon a GET request, the database may be accessed for some information that will be passed into a template.

As seen in Figure 4.2.1, templates are accessed by Jinja2, which is a popular library for templating files using Python. Virtually all web pages served by GeekSurvey are extended from `templates/base.html`, which contains a basic html `<head>` as well as a header and footer containing navigation links. Jinja2 is given access to some data through `views.py`. This data can be accessed during templating execution, and/or inserted into the final web page. For instance, if you are not signed in, the GeekSurvey home page may show a default page with a prompt to log in. If you are signed in, the same web page may show your username and some links to other parts of the web application.

5. Testing

Software testing is needed to ensure that GeekSurvey does what it is intended to do. Simply, software testing is the process of evaluating and verifying an application and its functionalities. The goal is to make sure that the application works as intended and that there are no bugs that went unnoticed. By testing software, performance can be improved and the development cost can be reduced since errors will be detected earlier on in a project. Overall, software testing will improve quality and understanding of the software, reduce costs, and help produce software that functions cleanly without problems.

GeekSurvey has been tested in a variety of ways. Unit testing has been conducted to detect any errors that may occur within each part of the codebase and determine the cause of those errors. Regarding GeekSurvey, the current codebase contains some elements that can be tested using unit testing, but should ideally be refactored to better accommodate unit testing in the future. Then, integration testing has been performed to determine how external platforms, such as Google Forms and the Paypal API, interact and exchange data with GeekSurvey. The goal is to see if any difficulties or problems arise when using the external platforms in relation to GeekSurvey. Finally, usability testing has been conducted to see how end users use GeekSurvey and its features. There are features of GeekSurvey, from account creation to study enrollment, that will need to be tested multiple times with various accounts to completely determine if there are any functionality issues.

The testing plan for GeekSurvey was designed for a web application that will facilitate numerous end users interacting with each other, as well as external platforms. The size of GeekSurvey's codebase is quite large and dividing them into parts will make it easier to detect errors and add new code in an organized way. The tests conducted on GeekSurvey will aid in producing a platform that will not only do what it is intended to do, but also enhance the scalability for future development and updates.

5.1 Unit Testing

For unit testing, the preferred way to write tests for Django is using the standard Python library unittest. Django also provides further testing subclasses under the Django test module, which extends the unittest library. The team utilized the Django test module to test functions in GeekSurvey, such as the `can_enroll()` function. This function takes in a user and a study object and returns true if they can enroll in a study. For the `can_enroll()` function, we used the Django test module to create 7 unit tests that asserted true or false depending on if the correct conditions were met.

5.2 Integration Testing

As GeekSurvey does not have explicit modules, traditional Integration testing methods and harnesses to test integration between GeekSurvey and Google Forms and PayPal could

not be used. Thus, successful integration between GeekSurvey and Google Forms and PayPal must be confirmed by manually testing both features through usability testing.

5.3 Usability Testing

Finally, for usability testing, the team outlined a variety of general tasks for test users to complete. The tasks were to be completed with as little direction from an interviewer as possible, as the team wanted to observe how test users intuitively interacted with GeekSurvey. The team conducted five interviews, which can be seen in the following subsections. So far, most of the feedback has been very positive. The team received recommendations and suggestions on how certain design elements can be improved, such as by adding certain buttons or redirects after completing forms on the site. The team also received feedback on features users would like to see. An example of this is that when you cash out funds from your account, you must cash out your entire balance. It was suggested that the team implement a way to select the amount of money to cash out.

5.3.1 Usability Test Results #1

Usability Test Subject Information

Record basic test subject information and demographics below. Make sure to highlight answers in yellow.

Subject Name: Samantha Rodriguez

Subject Age: 23

Subject Gender: Female

Subject Race/Ethnicity: Hispanic

Subject Occupation: Student

Subject Research/Participation Experience: Has research experience with two research teams, one dealing with Ecology and the other with Biology.

Subject Technical Background: Has experience using both R and Python for research related tasks, as well as capstone projects. They are a Data Science and Computer Science double major with a minor in mathematics.

Task #1 - Sign Up/Sign In

For this task, test users will be asked to sign up with either an email address or sign in with a social account like GitHub or Google. The test user's sign up/sign in preference and overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Sign Up/Sign-in Page

For this subtask, test users should be asked to navigate to the sign up/sign-in page from the landing page at geeksurvey.xyz. Record any comments below and highlight them in yellow.

Comments: They said the sign up/sign-in page was pretty easy to find.

Subtask #2 - Prompt User to Either Sign Up or Sign In

For this subtask, test users should be asked to either sign up with an email account or sign in with a social account. Please specify their selected choice and record any comments below. Make sure to highlight them in yellow.

Comments: The test user selected GitHub for sign in. They thought the process was fairly simple and straightforward, but they felt a bit uncomfortable allowing the GitHub token to have access to their GitHub account.

Task #2 - Edit User Profile

For this task, test users will edit their profile upon account creation and add any information they find relevant. The test user's overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Edit Profile Page

For this subtask, test users should be asked to navigate to the edit profile page from the signed-in landing page at geeksurvey.xyz. Record any comments below and highlight them in yellow.

Comments: They said it was easy to find but that the "Complete Profile" message was a bit confusing at first. They suggested that it might be more helpful if it just said "edit profile" from the start.

Subtask #2 - Prompt User to Edit and Complete their Profile

For this subtask, test users should be asked to complete their profile by editing all the fields in the edit profile form. Record any comments below and highlight them in yellow.

Comments: They said that in the profile update form there is no space between the footer and the submit/update button, and that it would be nice to add some space/padding.

Task #3 - Create a Study

For this task, test users will create a study object on the platform. This task will also direct test users to create a mock survey on Google Forms for use in the study object. The test user's overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Research Dashboard

For this subtask, test users should be asked to navigate to the research dashboard page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was easy to find the research dashboard.

Subtask #2 - Prompt User to Create a Study Object

For this subtask, test users should be asked to create a study object. As part of this subtask, direct the test user to create a Google Form that contains their chosen completion code. Also direct the test user to insert the link to their Google Form in the survey URL field of the study creation form. Make sure to also explain how the additional enrollment criteria are not necessary to fill out, but that they can be filled out if they choose. Record any comments below and highlight them in yellow.

Comments: They said the general workflow is pretty clear. They also said that the completion code text box being so big made it seem like a large code was needed, so maybe that can be changed.

Task #4 - Fund a Study

For this task, test users will first fund their accounts. In order to do this, the PayPal sandbox credentials will be shared with the test user. They will then return to their previously created study and add funds to it. The test user's experience with the PayPal payment system and with moving funds from their profile to the study object will be recorded.

Subtask #1 - Prompt User to Return to Profile Page

For this subtask, test users should be asked to navigate to the profile page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was easy to return to the profile page.

Subtask #2 - Prompt User to Add Funds to their Account

For this subtask, test users should be asked to add funds to their account. As part of this subtask, you must provide the test user with PayPal sandbox credentials. The credentials for the PayPal sandbox are:

Account Email: sb-1y5j013851137@personal.example.com

Password: W!T2i)N7

The test users should then be able to complete the PayPal transaction. Make sure to explain to the test users that the payment is being processed in a sandbox environment and that there are currently limitations in place, such as only being able to pay with the PayPal balance. Make sure to also mention that their account balance may not update immediately, as GeekSurvey is waiting on a response from PayPal. The test user may need to refresh a couple of times for the balance to update. Record any comments below and highlight them in yellow.

Comments: They said the process was straightforward. They appreciated the redirect back to the user profile after completing the entire process.

Subtask #3 - Prompt User to Return to Research Dashboard

For this subtask, test users should be asked to navigate back to the research dashboard from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was straightforward to return to the research dashboard.

Subtask #4 - Prompt User to Add Funds to Study

For this subtask, test users should be asked to fund their previously created study. Record any comments below and highlight them in yellow.

Comments: They said it would be nice to see what the compensation amount is when adding funds to a study. It would serve as a helpful reminder when loading funds.

Task #5 - Participate in a Study

For this task, test users will attempt to discover some pre-created studies. They will then enroll in a study and complete it. The test user's experience with the study participation workflow will be recorded.

Subtask #1 - Prompt User to Find Participation Dashboard

For this subtask, test users should be asked to navigate to the participation dashboard from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was easy to find the participation dashboard.

Subtask #2 - Prompt User to Discover New Studies

For this subtask, test users should be asked to discover new studies they are eligible to enroll in. Record any comments below and highlight them in yellow.

Comments: They said it was easy to discover new studies from the participation dashboard.

Subtask #3 - Prompt User to Note the Available Studies

For this subtask, it should be explained to test users that they can only see studies that they are eligible to participate in based on how they filled out their profile. Depending on how they edited their profile, they may see the "Strategies for Culminating an Open Source Community" study, the "Python vs. R for Data Science" study, a combination of the two, or no studies. Record any comments below and highlight them in yellow.

Comments: The test user was unable to see their own study in the study discovery menu despite their profile criteria matching. It is worth looking into the cause of this later. They said the layout was nice. They also said it would be nice if the view buttons for studies were aligned relative to adjacent studies. They also said it would also be nice if the whitespace on top of the studies was equal in height to the whitespace below the studies. They would also like for the compensation amount to be visible in the study discovery page.

Subtask #4 - Prompt User to Edit their Profile

For this subtask, the test users will be editing their profile to act as if they are a Data Scientist with at least one year of experience, a Bachelor's Degree, and have experience with open source development. Please ensure that these fields have been correctly updated by the test user, as these fields are important for the following subtasks. Record any comments below and highlight them in yellow.

Comments: They said it was pretty easy. They also said that It would be nice if there was uniformity between default options in drop downs, specifically regarding their presentation. They cited that some dropdowns defaulted to the top of the list, whereas others defaulted elsewhere. This would make it easier as someone who tabs through dropdowns.

Subtask #5 - Prompt User to Return to Study Discovery

For this subtask, the test users will be asked to navigate back to the study discovery page. Once here, they should note that editing their profile has caused them to be able to see the "Python vs. R for Data Science" study (unless they were already able to see it due to their previous profile configuration). Record any comments below and highlight them in yellow.

Comments: They said it was pretty straightforward.

Subtask #6 - Prompt User to Participate in Study

For this subtask, the test users will be asked to participate in the "Python vs. R for Data Science" study. As part of this subtask, direct the user to view the study and enroll in it. Once they are enrolled, direct the user to complete the external study and note the completion code. Finally, the user should be directed back to the study's completion page, where they can enter the completion code. Record any comments below and highlight them in yellow.

Comments: They said it would be nice for there to be a message saying that a study is completed after completing the study on the view page. Currently, it just shows some basic study information. Other than that, they said it was very straightforward.

Task #6 - Remove Funds from Account

For this task, test users will remove funds from their account after being compensated for completing the previous study. The test user's experience with removing funds from their account will be recorded.

Subtask #1 - Prompt User to Return to Profile Page

For this subtask, test users should be asked to navigate to the profile page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was straightforward.

Subtask #2 - Prompt User to Remove Funds from Account

For this subtask, test users should be asked to remove funds from their account. For this subtask, test users must use a "valid" PayPal email address to remove funds from their account.

Make sure to explain to the test users that this currently requires them to use the same sandbox credentials used earlier, as the payment processing system is operating in a sandbox environment. Once again, the email used for the PayPal Sandbox is:

Account Email: sb-1y5j013851137@personal.example.com

The test users should then be able to successfully remove funds from their account. Record any comments below and highlight them in yellow.

Comments: They said it would be nice to be able to select the amount of money to be withdrawn.

5.3.2 Usability Test Results #2

Usability Test Subject Information

Record basic test subject information and demographics below. Make sure to highlight answers in yellow.

Subject Name: Andres Jose Rodriguez

Subject Age: 21

Subject Gender: Male

Subject Race/Ethnicity: Hispanic

Subject Occupation: Research Assistant and Caregiver

Subject Research/Participation Experience: Has experience with microbiology research and biology research. Has also conducted a literature review against Oncolytic Adenoviruses and their benefits on Malignant Glioma.

Subject Technical Background: Has experience with real-time Polymerase Chain Reaction (rtPCR). Also has experience with MATLAB and MEGA7. They graduated in Fall of 2021 with a Bachelor's in Biomedical Sciences and minors in Chemistry and Psychology.

Task #1 - Sign Up/Sign In

For this task, test users will be asked to sign up with either an email address or sign in with a social account like GitHub or Google. The test user's sign up/sign in preference and overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Sign Up/Sign-in Page

For this subtask, test users should be asked to navigate to the sign up/sign-in page from the landing page at geeksurvey.xyz. Record any comments below and highlight them in yellow.

Comments: They said it was pretty easy to find.

Subtask #2 - Prompt User to Either Sign Up or Sign In

For this subtask, test users should be asked to either sign up with an email account or sign in with a social account. Please specify their selected choice and record any comments below. Make sure to highlight them in yellow.

Comments: The test user selected to sign up using an email. They thought the process was easy, but commented that their original password was rejected for being common. They would like to be prompted with examples of what they could add to their password to make it less common, such as specific characters.

Task #2 - Edit User Profile

For this task, test users will edit their profile upon account creation and add any information they find relevant. The test user's overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Edit Profile Page

For this subtask, test users should be asked to navigate to the edit profile page from the signed-in landing page at geeksurvey.xyz. Record any comments below and highlight them in yellow.

Comments: They said it was easy but that it would be nice if the purpose of the "Complete Profile" button was more clear when first editing the profile.

Subtask #2 - Prompt User to Edit and Complete their Profile

For this subtask, test users should be asked to complete their profile by editing all the fields in the edit profile form. Record any comments below and highlight them in yellow.

Comments: They said it was simple and straightforward, although they noted that the years of experience field is a bit too broad. They commented that it would be nice if there were more specific years of experience categories.

Task #3 - Create a Study

For this task, test users will create a study object on the platform. This task will also direct test users to create a mock survey on Google Forms for use in the study object. The test user's overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Research Dashboard

For this subtask, test users should be asked to navigate to the research dashboard page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was easy to find.

Subtask #2 - Prompt User to Create a Study Object

For this subtask, test users should be asked to create a study object. As part of this subtask, direct the test user to create a Google Form that contains their chosen completion code. Also direct the test user to insert the link to their Google Form in the survey URL field of the study creation form. Make sure to also explain how the additional enrollment criteria are not necessary to fill out, but that they can be filled out if they choose. Record any comments below and highlight them in yellow.

Comments: They said the process was pretty straightforward. They commented that the completion code box could be shrunk down a bit. They also commented that they liked the way the study edit form submission redirects you to the research dashboard and shows you your studies.

Task #4 - Fund a Study

For this task, test users will first fund their accounts. In order to do this, the PayPal sandbox credentials will be shared with the test user. They will then return to their previously created study and add funds to it. The test user's experience with the PayPal payment system and with moving funds from their profile to the study object will be recorded.

Subtask #1 - Prompt User to Return to Profile Page

For this subtask, test users should be asked to navigate to the profile page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was satisfactory and easy to do.

Subtask #2 - Prompt User to Add Funds to their Account

For this subtask, test users should be asked to add funds to their account. As part of this subtask, you must provide the test user with PayPal sandbox credentials. The credentials for the PayPal sandbox are:

Account Email: sb-1y5j013851137@personal.example.com

Password: W!T2i)N7

The test users should then be able to complete the PayPal transaction. Make sure to explain to the test users that the payment is being processed in a sandbox environment and that there are currently limitations in place, such as only being able to pay with the PayPal balance. Make sure to also mention that their account balance may not update immediately, as GeekSurvey is waiting on a response from PayPal. The test user may need to refresh a couple of times for the balance to update. Record any comments below and highlight them in yellow.

Comments: They said the process was straightforward. They also appreciated the redirect back to the profile. They liked that they could see the updated balance as well.

Subtask #3 - Prompt User to Return to Research Dashboard

For this subtask, test users should be asked to navigate back to the research dashboard from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was easy.

Subtask #4 - Prompt User to Add Funds to Study

For this subtask, test users should be asked to fund their previously created study. Record any comments below and highlight them in yellow.

Comments: They said it was straightforward. They appreciated the tip about negative amounts withdrawing funds from the study. They also commented that the process seems very easy and refined.

Task #5 - Participate in a Study

For this task, test users will attempt to discover some pre-created studies. They will then enroll in a study and complete it. The test user's experience with the study participation workflow will be recorded.

Subtask #1 - Prompt User to Find Participation Dashboard

For this subtask, test users should be asked to navigate to the participation dashboard from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was easy to find.

Subtask #2 - Prompt User to Discover New Studies

For this subtask, test users should be asked to discover new studies they are eligible to enroll in. Record any comments below and highlight them in yellow.

Comments: They said that it was fine and easy to do.

Subtask #3 - Prompt User to Note the Available Studies

For this subtask, it should be explained to test users that they can only see studies that they are eligible to participate in based on how they filled out their profile. Depending on how they edited their profile, they may see the "Strategies for Culminating an Open Source Community" study, the "Python vs. R for Data Science" study, a combination of the two, or no studies. Record any comments below and highlight them in yellow.

Comments: They liked how the studies were presented. They said they would like for the compensation amount to be visible in the study discovery page.

Subtask #4 - Prompt User to Edit their Profile

For this subtask, the test users will be editing their profile to act as if they are a Data Scientist with at least one year of experience, a Bachelor's Degree, and have experience with open source development. Please ensure that these fields have been correctly updated by the test user, as these fields are important for the following subtasks. Record any comments below and highlight them in yellow.

Comments: They said that it would be nice for the dropdown fields to be in alphabetical order.

Subtask #5 - Prompt User to Return to Study Discovery

For this subtask, the test users will be asked to navigate back to the study discovery page. Once here, they should note that editing their profile has caused them to be able to see the “Python vs. R for Data Science” study (unless they were already able to see it due to their previous profile configuration). Record any comments below and highlight them in yellow.

Comments: They said it was easy.

Subtask #6 - Prompt User to Participate in Study

For this subtask, the test users will be asked to participate in the “Python vs. R for Data Science” study. As part of this subtask, direct the user to view the study and enroll in it. Once they are enrolled, direct the user to complete the external study and note the completion code. Finally, the user should be directed back to the study’s completion page, where they can enter the completion code. Record any comments below and highlight them in yellow.

Comments: They said it would be nice to mention in the help page that you need to note the completion code when taking the survey.

Task #6 - Remove Funds from Account

For this task, test users will remove funds from their account after being compensated for completing the previous study. The test user’s experience with removing funds from their account will be recorded.

Subtask #1 - Prompt User to Return to Profile Page

For this subtask, test users should be asked to navigate to the profile page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was easy.

Subtask #2 - Prompt User to Remove Funds from Account

For this subtask, test users should be asked to remove funds from their account. For this subtask, test users must use a “valid” PayPal email address to remove funds from their account. Make sure to explain to the test users that this currently requires them to use the same sandbox credentials used earlier, as the payment processing system is operating in a sandbox environment. Once again, the email used for the PayPal Sandbox is:

Account Email: sb-1y5j013851137@personal.example.com

The test users should then be able to successfully remove funds from their account. Record any comments below and highlight them in yellow.

Comments: They said it would be nice if you could say how much you want to claim. They also said it would be nice if there was a dropdown that increments in \$25 chunks. They also

suggested that one way to reduce the number of transactions/claims would be to incentivize users to cash out less frequently by offering a bonus for larger amounts.

5.3.3 Usability Test Results #3

Usability Test Subject Information

Record basic test subject information and demographics below. Make sure to highlight answers in yellow.

Subject Name: Randy Salas

Subject Age: 22

Subject Gender: Male

Subject Race/Ethnicity: Hispanic

Subject Occupation: Student

Subject Research/Participation Experience: The test subject has experience with lab research on physiological effects of taste on millipedes using salt and sugar vegetable jelly.

Subject Technical Background: Biology

Task #1 - Sign Up/Sign In

For this task, test users will be asked to sign up with either an email address or sign in with a social account like GitHub or Google. The test user's sign up/sign in preference and overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Sign Up/Sign-in Page

For this subtask, test users should be asked to navigate to the sign up/sign-in page from the landing page at geeksurvey.xyz. Record any comments below and highlight them in yellow.

Comments: They said it was a very elegant and simple user interface when signing up, and also a very easy process.

Subtask #2 - Prompt User to Either Sign Up or Sign In

For this subtask, test users should be asked to either sign up with an email account or sign in with a social account. Please specify their selected choice and record any comments below. Make sure to highlight them in yellow.

Comments: The test user selected to sign up using an email. They said it was very easy to sign up and that it was a quick process.

Task #2 - Edit User Profile

For this task, test users will edit their profile upon account creation and add any information they find relevant. The test user's overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Edit Profile Page

For this subtask, test users should be asked to navigate to the edit profile page from the signed-in landing page at geeksurvey.xyz. Record any comments below and highlight them in yellow.

Comments: They said it was very easy to find as there is a notification that pops up to finish the process.

Subtask #2 - Prompt User to Edit and Complete their Profile

For this subtask, test users should be asked to complete their profile by editing all the fields in the edit profile form. Record any comments below and highlight them in yellow.

Comments: They said it was a simple layout and that the pop up scroll menus make it easier to fill out.

Task #3 - Create a Study

For this task, test users will create a study object on the platform. This task will also direct test users to create a mock survey on Google Forms for use in the study object. The test user's overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Research Dashboard

For this subtask, test users should be asked to navigate to the research dashboard page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said that it was simple and that there is a noticeable button to get back.

Subtask #2 - Prompt User to Create a Study Object

For this subtask, test users should be asked to create a study object. As part of this subtask, direct the test user to create a Google Form that contains their chosen completion code. Also direct the test user to insert the link to their Google Form in the survey URL field of the study creation form. Make sure to also explain how the additional enrollment criteria are not necessary to fill out, but that they can be filled out if they choose. Record any comments below and highlight them in yellow.

Comments: They said it was a very simple process to create a survey and study from start to finish in just minutes.

Task #4 - Fund a Study

For this task, test users will first fund their accounts. In order to do this, the PayPal sandbox credentials will be shared with the test user. They will then return to their previously

created study and add funds to it. The test user's experience with the PayPal payment system and with moving funds from their profile to the study object will be recorded.

Subtask #1 - Prompt User to Return to Profile Page

For this subtask, test users should be asked to navigate to the profile page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was easy and simple to find.

Subtask #2 - Prompt User to Add Funds to their Account

For this subtask, test users should be asked to add funds to their account. As part of this subtask, you must provide the test user with PayPal sandbox credentials. The credentials for the PayPal sandbox are:

Account Email: sb-1y5j013851137@personal.example.com

Password: W!T2i)N7

The test users should then be able to complete the PayPal transaction. Make sure to explain to the test users that the payment is being processed in a sandbox environment and that there are currently limitations in place, such as only being able to pay with the PayPal balance. Make sure to also mention that their account balance may not update immediately, as GeekSurvey is waiting on a response from PayPal. The test user may need to refresh a couple of times for the balance to update. Record any comments below and highlight them in yellow.

Comments: They said that the payment made towards their survey was very quickly added into their account simultaneously.

Subtask #3 - Prompt User to Return to Research Dashboard

For this subtask, test users should be asked to navigate back to the research dashboard from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was easy to find.

Subtask #4 - Prompt User to Add Funds to Study

For this subtask, test users should be asked to fund their previously created study. Record any comments below and highlight them in yellow.

Comments: The test user said it was a quick and easy process.

Task #5 - Participate in a Study

For this task, test users will attempt to discover some pre-created studies. They will then enroll in a study and complete it. The test user's experience with the study participation workflow will be recorded.

Subtask #1 - Prompt User to Find Participation Dashboard

For this subtask, test users should be asked to navigate to the participation dashboard from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was easy to find and get to.

Subtask #2 - Prompt User to Discover New Studies

For this subtask, test users should be asked to discover new studies they are eligible to enroll in. Record any comments below and highlight them in yellow.

Comments: They said that there were currently no studies to participate in but that it was a very simple process to get there and that the big blue button makes it easy to find where to go.

Subtask #3 - Prompt User to Note the Available Studies

For this subtask, it should be explained to test users that they can only see studies that they are eligible to participate in based on how they filled out their profile. Depending on how they edited their profile, they may see the “Strategies for Culminating an Open Source Community” study, the “Python vs. R for Data Science” study, a combination of the two, or no studies. Record any comments below and highlight them in yellow.

Comments: They said that they did not see any studies for the profile they made.

Subtask #4 - Prompt User to Edit their Profile

For this subtask, the test users will be editing their profile to act as if they are a Data Scientist with at least one year of experience, a Bachelor’s Degree, and have experience with open source development. Please ensure that these fields have been correctly updated by the test user, as these fields are important for the following subtasks. Record any comments below and highlight them in yellow.

Comments: They said that the fields were easily changed.

Subtask #5 - Prompt User to Return to Study Discovery

For this subtask, the test users will be asked to navigate back to the study discovery page. Once here, they should note that editing their profile has caused them to be able to see the “Python vs. R for Data Science” study (unless they were already able to see it due to their previous profile configuration). Record any comments below and highlight them in yellow.

Comments: The test user said that both pre-made studies popped up immediately after editing their profile.

Subtask #6 - Prompt User to Participate in Study

For this subtask, the test users will be asked to participate in the “Python vs. R for Data Science” study. As part of this subtask, direct the user to view the study and enroll in it. Once they are enrolled, direct the user to complete the external study and note the completion code.

Finally, the user should be directed back to the study's completion page, where they can enter the completion code. Record any comments below and highlight them in yellow.

Comments: They said that being given a completion code after finishing a survey was a very efficient way to make sure the participant completes the study. They also stated that using a Google Forms survey isn't the most efficient way to conduct studies, and that something within the website itself would be ideal.

Task #6 - Remove Funds from Account

For this task, test users will remove funds from their account after being compensated for completing the previous study. The test user's experience with removing funds from their account will be recorded.

Subtask #1 - Prompt User to Return to Profile Page

For this subtask, test users should be asked to navigate to the profile page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: The test user had no difficulties navigating to their profile from anywhere on the site.

Subtask #2 - Prompt User to Remove Funds from Account

For this subtask, test users should be asked to remove funds from their account. For this subtask, test users must use a "valid" PayPal email address to remove funds from their account. Make sure to explain to the test users that this currently requires them to use the same sandbox credentials used earlier, as the payment processing system is operating in a sandbox environment. Once again, the email used for the PayPal Sandbox is:

Account Email: sb-1y5j013851137@personal.example.com

The test users should then be able to successfully remove funds from their account. Record any comments below and highlight them in yellow.

Comments: They said that the money was easily removed after selecting the remove funds tab.

5.3.4 Usability Test Results #4

Usability Test Subject Information

Record basic test subject information and demographics below. Make sure to highlight answers in yellow.

Subject Name: Jiasheng Yang

Subject Age: 22

Subject Gender: Male

Subject Race/Ethnicity: Asian

Subject Occupation: Student

Subject Research/Participation Experience: Has research experience with one research team, dealing with medical image detection.

Subject Technical Background: They are majoring in Computer Science, can use python and R professionally, and have participated in a capstone project about artificial intelligence.

Task #1 - Sign Up/Sign In

For this task, test users will be asked to sign up with either an email address or sign in with a social account like GitHub or Google. The test user's sign up/sign in preference and overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Sign Up/Sign-in Page

For this subtask, test users should be asked to navigate to the sign up/sign-in page from the landing page at geeksurvey.xyz. Record any comments below and highlight them in yellow.

Comments: The test user said the Sign Up/Sign in page is easy to find. There is an obvious "Get Started" button, when I click the button, I will be redirected to the Sign Up/Sign in page.

Subtask #2 - Prompt User to Either Sign Up or Sign In

For this subtask, test users should be asked to either sign up with an email account or sign in with a social account. Please specify their selected choice and record any comments below. Make sure to highlight them in yellow.

Comments: The test user selected to sign up with an email account. He said it is quite easy to create an account for GeekSurvey with an email account.

Task #2 - Edit User Profile

For this task, test users will edit their profile upon account creation and add any information they find relevant. The test user's overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Edit Profile Page

For this subtask, test users should be asked to navigate to the edit profile page from the signed-in landing page at geeksurvey.xyz. Record any comments below and highlight them in yellow.

Comments: The test user said it was easy to find the "Profile Page" and edit it because there is a notification on the user name. When this notification was clicked on, they found the Complete Profile Page button.

Subtask #2 - Prompt User to Edit and Complete their Profile

For this subtask, test users should be asked to complete their profile by editing all the fields in the edit profile form. Record any comments below and highlight them in yellow.

Comments: The test user said that the profile fields were very comprehensive. They also said all the fields were mandatory to fill, which made them a bit uncomfortable. Also, for the "Bio" field, it seemed that there was a word limit, which they suggested would be nice if the word restriction was relaxed.

Task #3 - Create a Study

For this task, test users will create a study object on the platform. This task will also direct test users to create a mock survey on Google Forms for use in the study object. The test user's overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Research Dashboard

For this subtask, test users should be asked to navigate to the research dashboard page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: The test user said it was easy to find the research dashboard.

Subtask #2 - Prompt User to Create a Study Object

For this subtask, test users should be asked to create a study object. As part of this subtask, direct the test user to create a Google Form that contains their chosen completion code. Also direct the test user to insert the link to their Google Form in the survey URL field of the study creation form. Make sure to also explain how the additional enrollment criteria are not necessary to fill out, but that they can be filled out if they choose. Record any comments below and highlight them in yellow.

Comments: The test user said it is straightforward to create a study. They did, however, find the field "Survey URL" to be a bit confusing. They suggested that it would be better to have a short explanation for the "Survey URL", like changing the field name into "Google Form Survey URL."

Task #4 - Fund a Study

For this task, test users will first fund their accounts. In order to do this, the PayPal sandbox credentials will be shared with the test user. They will then return to their previously created study and add funds to it. The test user's experience with the PayPal payment system and with moving funds from their profile to the study object will be recorded.

Subtask #1 - Prompt User to Return to Profile Page

For this subtask, test users should be asked to navigate to the profile page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: The test user said it was easy to navigate to the profile page.

Subtask #2 - Prompt User to Add Funds to their Account

For this subtask, test users should be asked to add funds to their account. As part of this subtask, you must provide the test user with PayPal sandbox credentials. The credentials for the PayPal sandbox are:

Account Email: sb-1y5j013851137@personal.example.com

Password: W!T2i)N7

The test users should then be able to complete the PayPal transaction. Make sure to explain to the test users that the payment is being processed in a sandbox environment and that there are currently limitations in place, such as only being able to pay with the PayPal balance. Make sure to also mention that their account balance may not update immediately, as GeekSurvey is waiting on a response from PayPal. The test user may need to refresh a couple of times for the balance to update. Record any comments below and highlight them in yellow.

Comments: The test user said it was easy to purchase funds from PayPal and that it was really nice to redirect back to the profile.

Subtask #3 - Prompt User to Return to Research Dashboard

For this subtask, test users should be asked to navigate back to the research dashboard from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: The test user said it was straightforward to navigate back to the research dashboard.

Subtask #4 - Prompt User to Add Funds to Study

For this subtask, test users should be asked to fund their previously created study. Record any comments below and highlight them in yellow.

Comments: They said it was amazing to see the transfer from the account balance to the study compensation. They also suggested that it would be better to have a button to navigate back to the research dashboard after adding funds.

Task #5 - Participate in a Study

For this task, test users will attempt to discover some pre-created studies. They will then enroll in a study and complete it. The test user's experience with the study participation workflow will be recorded.

Subtask #1 - Prompt User to Find Participation Dashboard

For this subtask, test users should be asked to navigate to the participation dashboard from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was easy to find the participation dashboard.

Subtask #2 - Prompt User to Discover New Studies

For this subtask, test users should be asked to discover new studies they are eligible to enroll in. Record any comments below and highlight them in yellow.

Comments: They found three studies they can participate in, including the one he created. They were concerned about how much compensation was left, which means he wants to see the compensation on every study landing page.

Subtask #3 - Prompt User to Note the Available Studies

For this subtask, it should be explained to test users that they can only see studies that they are eligible to participate in based on how they filled out their profile. Depending on how they edited their profile, they may see the “Strategies for Culminating an Open Source Community” study, the “Python vs. R for Data Science” study, a combination of the two, or no studies. Record any comments below and highlight them in yellow.

Comments: The test user said it was amazing to see that when they changed their profile to have no experience of open source, they were no longer able to discover the study “Strategies for Culminating an Open Source Community” anymore. It was pretty nice.

Subtask #4 - Prompt User to Edit their Profile

For this subtask, the test users will be editing their profile to act as if they are a Data Scientist with at least one year of experience, a Bachelor’s Degree, and have experience with open source development. Please ensure that these fields have been correctly updated by the test user, as these fields are important for the following subtasks. Record any comments below and highlight them in yellow.

Comments: They said it was easy to update their profile.

Subtask #5 - Prompt User to Return to Study Discovery

For this subtask, the test users will be asked to navigate back to the study discovery page. Once here, they should note that editing their profile has caused them to be able to see the “Python vs. R for Data Science” study (unless they were already able to see it due to their previous profile configuration). Record any comments below and highlight them in yellow.

Comments: They said yes, they can now see the “Python vs. R for Data Science” study which they could not initially.

Subtask #6 - Prompt User to Participate in Study

For this subtask, the test users will be asked to participate in the “Python vs. R for Data Science” study. As part of this subtask, direct the user to view the study and enroll in it. Once they are enrolled, direct the user to complete the external study and note the completion code. Finally, the user should be directed back to the study’s completion page, where they can enter the completion code. Record any comments below and highlight them in yellow.

Comments: They said it was straightforward to follow the workflow, but that it would be better to highlight the secret code.

Task #6 - Remove Funds from Account

For this task, test users will remove funds from their account after being compensated for completing the previous study. The test user's experience with removing funds from their account will be recorded.

Subtask #1 - Prompt User to Return to Profile Page

For this subtask, test users should be asked to navigate to the profile page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said that it was pretty easy to navigate to the profile page.

Subtask #2 - Prompt User to Remove Funds from Account

For this subtask, test users should be asked to remove funds from their account. For this subtask, test users must use a "valid" PayPal email address to remove funds from their account. Make sure to explain to the test users that this currently requires them to use the same sandbox credentials used earlier, as the payment processing system is operating in a sandbox environment. Once again, the email used for the PayPal Sandbox is:

Account Email: sb-1y5j013851137@personal.example.com

The test users should then be able to successfully remove funds from their account. Record any comments below and highlight them in yellow.

Comments: They said that because it was so simple to withdraw money from their GeekSurvey account, it may be somewhat unsafe. For example, if one has entered the wrong account email, then the money will not come to their account, but rather the account associated with the wrong email.

5.3.5 Usability Test Results #5

Usability Test Subject Information

Record basic test subject information and demographics below. Make sure to highlight answers in yellow.

Subject Name: Jacob Penney

Subject Age: 29

Subject Gender: Non-binary

Subject Race/Ethnicity: White/Caucasian

Subject Occupation: Student and University Researcher

Subject Research/Participation Experience: The test subject has experience with working under the direction of a PhD student and fulfilling software tasks that they asked them to perform.

Subject Technical Background: Has experience with the fields of Computer Science and Cybersecurity from their studies at Northern Arizona University.

Task #1 - Sign Up/Sign In

For this task, test users will be asked to sign up with either an email address or sign in with a social account like GitHub or Google. The test user's sign up/sign in preference and overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Sign Up/Sign-in Page

For this subtask, test users should be asked to navigate to the sign up/sign-in page from the landing page at geeksurvey.xyz. Record any comments below and highlight them in yellow.

Comments: They said it was self-explanatory. The test user suggested adding the phrase "sign-up" to make it clearer.

Subtask #2 - Prompt User to Either Sign Up or Sign In

For this subtask, test users should be asked to either sign up with an email account or sign in with a social account. Please specify their selected choice and record any comments below. Make sure to highlight them in yellow.

Comments: The test user decided to create an account using an email address. They appreciated that the form was simple. They also said it was quick and straightforward.

Task #2 - Edit User Profile

For this task, test users will edit their profile upon account creation and add any information they find relevant. The test user's overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Edit Profile Page

For this subtask, test users should be asked to navigate to the edit profile page from the signed-in landing page at geeksurvey.xyz. Record any comments below and highlight them in yellow.

Comments: They said it was really easy to find. They suggested it would be nice to have something like a drop down menu for editing your profile.

Subtask #2 - Prompt User to Edit and Complete their Profile

For this subtask, test users should be asked to complete their profile by editing all the fields in the edit profile form. Record any comments below and highlight them in yellow.

Comments: They felt like the edit profile form fit their needs and was not too verbose. It felt appropriate. There was nothing flashy or distracting.

Task #3 - Create a Study

For this task, test users will create a study object on the platform. This task will also direct test users to create a mock survey on Google Forms for use in the study object. The test user's overall experience with this task will be recorded.

Subtask #1 - Prompt User to Find the Research Dashboard

For this subtask, test users should be asked to navigate to the research dashboard page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was pretty clear. The first thing that they noticed was that there was both a "research" and "participate" link, which made it clear that these were binary options. Since research can be both a noun and a verb, having "participate" next to it made it clear that they were both action links.

Subtask #2 - Prompt User to Create a Study Object

For this subtask, test users should be asked to create a study object. As part of this subtask, direct the test user to create a Google Form that contains their chosen completion code. Also direct the test user to insert the link to their Google Form in the survey URL field of the study creation form. Make sure to also explain how the additional enrollment criteria are not necessary to fill out, but that they can be filled out if they choose. Record any comments below and highlight them in yellow.

Comments: They really enjoyed the boxes in the study form that provided ample space to fill out. They did not like the enrollment criteria part of the form as it was quite condensed and hard to look at. There were too many lines that made it seem a bit inaccessible. It would also be nice for there to be more padding for the submit button. The general workflow for creating the study and implementing the Google Form was pretty straightforward, however it would be nice for fields like the completion code to have some kind of explanation as the missing context could make it difficult to figure out. They had an overall positive impression of the research dashboard.

Task #4 - Fund a Study

For this task, test users will first fund their accounts. In order to do this, the PayPal sandbox credentials will be shared with the test user. They will then return to their previously created study and add funds to it. The test user's experience with the PayPal payment system and with moving funds from their profile to the study object will be recorded.

Subtask #1 - Prompt User to Return to Profile Page

For this subtask, test users should be asked to navigate to the profile page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was pretty logical.

Subtask #2 - Prompt User to Add Funds to their Account

For this subtask, test users should be asked to add funds to their account. As part of this subtask, you must provide the test user with PayPal sandbox credentials. The credentials for the PayPal sandbox are:

Account Email: sb-1y5j013851137@personal.example.com

Password: W!T2i)N7

The test users should then be able to complete the PayPal transaction. Make sure to explain to the test users that the payment is being processed in a sandbox environment and that there are currently limitations in place, such as only being able to pay with the PayPal balance. Make sure to also mention that their account balance may not update immediately, as GeekSurvey is waiting on a response from PayPal. The test user may need to refresh a couple of times for the balance to update. Record any comments below and highlight them in yellow.

Comments: They said the workflow was intuitive and not overly complicated. They appreciated the message for saying that there was a premium on purchasing funds. They would like to see more transparency on how the extra cost is used. They also felt that the initial add funds page could look more professional and legitimate.

Subtask #3 - Prompt User to Return to Research Dashboard

For this subtask, test users should be asked to navigate back to the research dashboard from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was straightforward, very clear, and accessible.

Subtask #4 - Prompt User to Add Funds to Study

For this subtask, test users should be asked to fund their previously created study. Record any comments below and highlight them in yellow.

Comments: They said it was a pretty straightforward process. They also said it is “no frills” and gets the job done.

Task #5 - Participate in a Study

For this task, test users will attempt to discover some pre-created studies. They will then enroll in a study and complete it. The test user’s experience with the study participation workflow will be recorded.

Subtask #1 - Prompt User to Find Participation Dashboard

For this subtask, test users should be asked to navigate to the participation dashboard from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was pretty clear.

Subtask #2 - Prompt User to Discover New Studies

For this subtask, test users should be asked to discover new studies they are eligible to enroll in. Record any comments below and highlight them in yellow.

Comments: They said it was very clear as to how to find and discover new studies.

Subtask #3 - Prompt User to Note the Available Studies

For this subtask, it should be explained to test users that they can only see studies that they are eligible to participate in based on how they filled out their profile. Depending on how they edited their profile, they may see the “Strategies for Culminating an Open Source Community” study, the “Python vs. R for Data Science” study, a combination of the two, or no studies. Record any comments below and highlight them in yellow.

Comments: They said it totally made sense.

Subtask #4 - Prompt User to Edit their Profile

For this subtask, the test users will be editing their profile to act as if they are a Data Scientist with at least one year of experience, a Bachelor’s Degree, and have experience with open source development. Please ensure that these fields have been correctly updated by the test user, as these fields are important for the following subtasks. Record any comments below and highlight them in yellow.

Comments: They said it was easy and clear.

Subtask #5 - Prompt User to Return to Study Discovery

For this subtask, the test users will be asked to navigate back to the study discovery page. Once here, they should note that editing their profile has caused them to be able to see the “Python vs. R for Data Science” study (unless they were already able to see it due to their previous profile configuration). Record any comments below and highlight them in yellow.

Comments: They said it made total sense. One thing they suggested is to add profile options on the study discovery page for quick and easy changes to your profile without leaving the study discovery page.

Subtask #6 - Prompt User to Participate in Study

For this subtask, the test users will be asked to participate in the “Python vs. R for Data Science” study. As part of this subtask, direct the user to view the study and enroll in it. Once they are enrolled, direct the user to complete the external study and note the completion code. Finally, the user should be directed back to the study’s completion page, where they can enter the completion code. Record any comments below and highlight them in yellow.

Comments: They said that participating in studies was really straightforward. There was nothing about them that seemed like nonsense. They felt positively about it.

Task #6 - Remove Funds from Account

For this task, test users will remove funds from their account after being compensated for completing the previous study. The test user's experience with removing funds from their account will be recorded.

Subtask #1 - Prompt User to Return to Profile Page

For this subtask, test users should be asked to navigate to the profile page from practically anywhere on the site. Record any comments below and highlight them in yellow.

Comments: They said it was pretty straightforward.

Subtask #2 - Prompt User to Remove Funds from Account

For this subtask, test users should be asked to remove funds from their account. For this subtask, test users must use a "valid" PayPal email address to remove funds from their account. Make sure to explain to the test users that this currently requires them to use the same sandbox credentials used earlier, as the payment processing system is operating in a sandbox environment. Once again, the email used for the PayPal Sandbox is:

Account Email: sb-1y5j013851137@personal.example.com

The test users should then be able to successfully remove funds from their account. Record any comments below and highlight them in yellow.

Comments: They said that removing funds or "cashing out" is straightforward. The user interface was clear and easy to understand.

6. Project Timeline

The legend for the team's Gantt Chart can be seen in Figure 6.1. A plan for development and a project timeline were necessary to ensure the completion of core requirements. The team spent the entire semester thus far implementing and completing core features as requested by the clients. Outside of core features, the team also continued to test and develop stretch goal features.






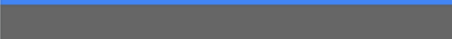
Legend	
Done	
TODO	
Due	
Winter Break	
Spring Break	
Now Line (NL)	
Class Deliverables	<i>Red Text and Italicized and Bolded</i>

Figure 6.1 - Gantt Chart Legend

As shown in Figure 6.2, the team has completed all the core requirements along with a couple of stretch goals for Geek Survey during the Spring 2022 semester. This was accomplished through eight required sprints, and an optional sprint with stretch goals. The eight required sprints were either completed one at a time or in parallel. Some sprints were also completed by splitting the team into programming pairs. Last Fall, the team planned for development and wrote the requirements for the project. The technologies used were determined and proven to be feasible. All documentation was finished and signed off by the clients. Our current progress is great. We have a dynamic and modern site with a Bootstrap front end and a Django backend.

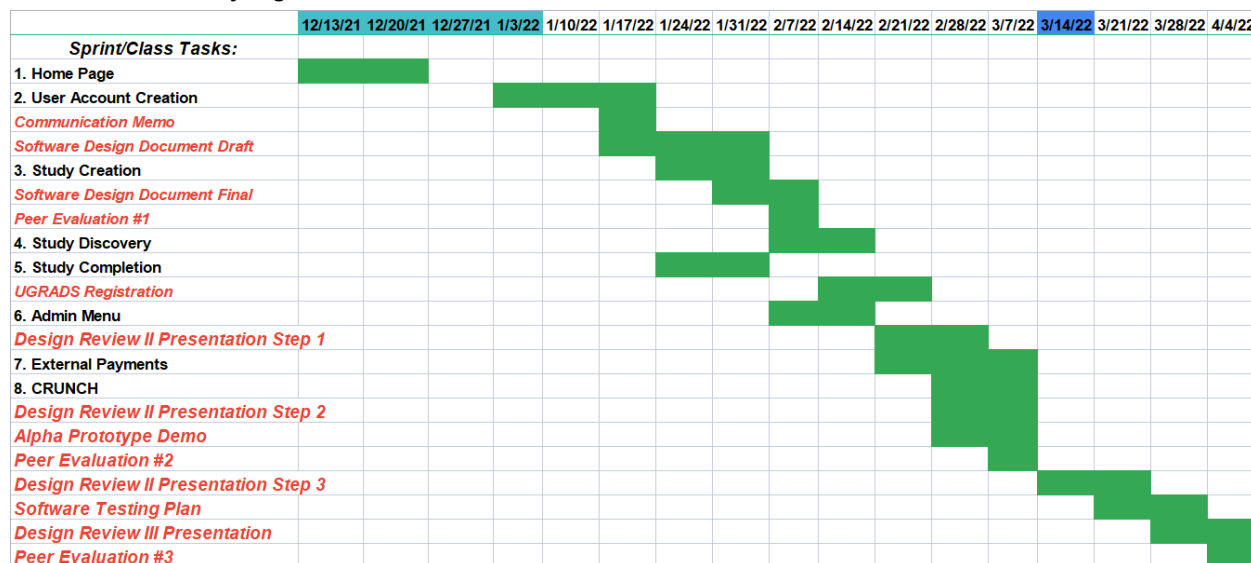


Figure 6.2 - Gantt Chart

7. Future Work

GeekSurvey is a fully featured web application deployed as a live development website. It includes many different web pages for managing profiles, studies, account funds, and more. The domain name and live server have been transferred to the clients for future work.

In the future, GeekSurvey could be refactored into a REST API, with the frontend being rewritten as a React.js application. It could be generalized to support more types of research. This would position GeekSurvey to be instanced by universities and non-profit organizations for their unique needs.

Currently, GeekSurvey uses a Paypal Sandbox for payments. It could be updated to use real payments after a security audit and improved business and legal compliance.

8. Conclusion

GeekSurvey is a web application to solve the coordination problems inherent to survey-based research. The core audience for GeekSurvey is researchers who need to recruit research participants with specialized profiles. The clients' business process is improved upon the deployment of GeekSurvey as it makes selective recruitment of qualified participants more efficient.

GeekSurvey has all of the core features needed to recruit for, manage, and actually conduct research studies. It is freely and publicly available, open source, easily extensible, and simple to deploy to any hosting platform. Therefore, GeekSurvey has the potential to bring serious value to any researchers in the world who wish to customize and improve their study recruitment workflow.

Appendix A: Development Environment and Toolchain

Hardware

The SuperGeeks team developed GeekSurvey on both Windows 10 and macOS Monterey. The technical specifications of each of the team members' development machines varied greatly. With that said, GeekSurvey does not require an intense amount of resources to run and/or develop. Some potential minimum requirements include a 7th generation Intel Core i3 CPU or 1st generation AMD Ryzen 3 CPU and 4GB of RAM. Some potential recommended requirements include a 7th generation Intel Core i5 CPU or 1st generation AMD Ryzen 5 CPU and 8GB of RAM.

Toolchain

The SuperGeeks team utilized a variety of development environments. For code editing, some team members elected to use text editors while others used IDEs. Some of the text editors used included Atom and Vim. One of the IDEs used was PyCharm. For testing and running changes, team members ran the GeekSurvey Django application locally through a command line or terminal. For git, team members utilized a command line interface, either through a native terminal/command prompt or through other applications, such as Git Bash. The team also elected to use SQLite for GeekSurvey's backend database, as it has simple-to-configure integration with Python and Django applications.

The SuperGeeks team also utilized other supportive tools and packages during development. First and foremost, the team used Python's standard package manager, pip, to install required packages and libraries for development. Some notable packages installed included `django-allauth`, `django-crispy-forms`, `crispy_bootstrap5`, `django-countries`, `django-paypal`, and `pytz`. The `django-allauth` package was used to handle the account creation and management for GeekSurvey. The `django-crispy-forms` and `crispy_bootstrap5` packages were used to integrate with the many forms that GeekSurvey uses to give them a more modern look and simple integration with the SQLite database. The `django-countries` package was used to include countries as options to form responses, which saved the team from having to insert hundreds of country options in the forms manually. The `django-paypal` package provided numerous options for interacting with the PayPal API for payments. Finally, the `pytz` package was used for handling timezones on the backend and in forms.

Setup

The following instructions can be followed to setup and deploy an instance of GeekSurvey either locally or to a server.

Prerequisites

Must have a machine with git, python3, and pip already installed.

Installing

1. `git clone https://github.com/NAU-SuperGeeks/geeksurvey.git && cd geeksurvey`
2. `pip install requirements.txt`
3. `python manage.py makemigrations geeksurvey`
4. `python manage.py migrate`

Testing

`python manage.py test`

Deployment

Deploy Locally

For a simple locally deployed version of GeekSurvey, do

1. `python manage.py runserver`
2. Go to `localhost:8000` in the web browser.

Features that integrate with other web APIs typically do not work when geeksurvey is deployed locally. This includes OAuth by Github and Google, PayPal payments, and email services.

Typically, various credentials for these APIs are stored in a `.env` file at the root of the project. These can be left out for local testing

Deploy to a Server

To deploy GeekSurvey to a server, a `.env` file must be in the project root containing:

1. A valid Django secret key used for encryption

```
GEEKSURVEY_SECRET_KEY = "some_data"
```

2. A debug flag set to false (0) for deployment

```
GEEKSURVEY_DEBUG = 0
```

3. Valid Gmail App credentials for email

```
GMAIL_APP_USERNAME = "user@gmail.com"
```

```
GMAIL_APP_PASSWORD = "sample-app-password"
```

4. Valid Google App credentials for OAuth

```
GOOGLE_AUTH_CLIENT = "some_data"
```

```
GOOGLE_AUTH_SECRET = "some_data"
```

5. Valid GitHub App credentials for OAuth

```
GITHUB_AUTH_CLIENT = "some_data"
```

```
GITHUB_AUTH_SECRET = "some_data"
```

6. Valid PayPal Developer credentials for all payments

```
PAYPAL_CLIENT_ID = "some_data"
```

```
PAYPAL_CLIENT_SECRET = "some_data"
```

6. Valid PayPal Business account credentials for paying out to users

```
PAYPAL_BIZ_ACCOUNT = "user@example.com"
```

GeekSurvey can be published on the web from a linux server by configuring nginx and gunicorn to run the project through a WSGI. A good reference for working through this process can be found [here](#).

Take care to host the static files separately to support caching. This requires python manage.py collectstatic to collect the static files into a single folder.

GeekSurvey will fail if it is not hosted with HTTPS, so make sure your domain name is certified and your nginx configuration uses SSL.