

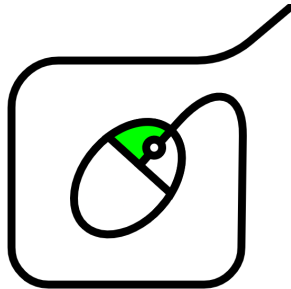
# Software Design Document Version 1.0

09 Feb 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

<b>1. Introduction</b>	<b>2</b>
<b>2. Implementation Overview</b>	<b>4</b>
<b>3. Architecture Overview</b>	<b>5</b>
<b>4. Module Descriptions</b>	<b>7</b>
<b>4.1 Templates</b>	<b>8</b>
<b>4.2 User Accounts</b>	<b>10</b>
<b>4.3 Studies</b>	<b>12</b>
<b>5. Implementation Plan</b>	<b>14</b>
<b>6. Conclusion</b>	<b>15</b>
<b>7. Glossary</b>	<b>16</b>
<b>8. Appendix: SuperGeeks Gantt Chart</b>	<b>17</b>

# 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 will serve as that product. It will be a public web application in which users are able to sign up to create an account and create, manage, or participate in research studies depending on the type of account they create. GeekSurvey will need core features to facilitate researchers and participants coordination such as:

- Two account types: researchers and participants. By default, all accounts can access researcher and participant features. These two account types are an abstraction to clarify the purpose of different GeekSurvey features. Accounts can be manually given administrative access.
- Participant accounts will include 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 will be 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. Implementation Overview

GeekSurvey will be 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 will have an associated Profile which can be edited by the user. This Profile will contain 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 can be checked programmatically when serving a Study enrollment POST request, because the backend will have 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.

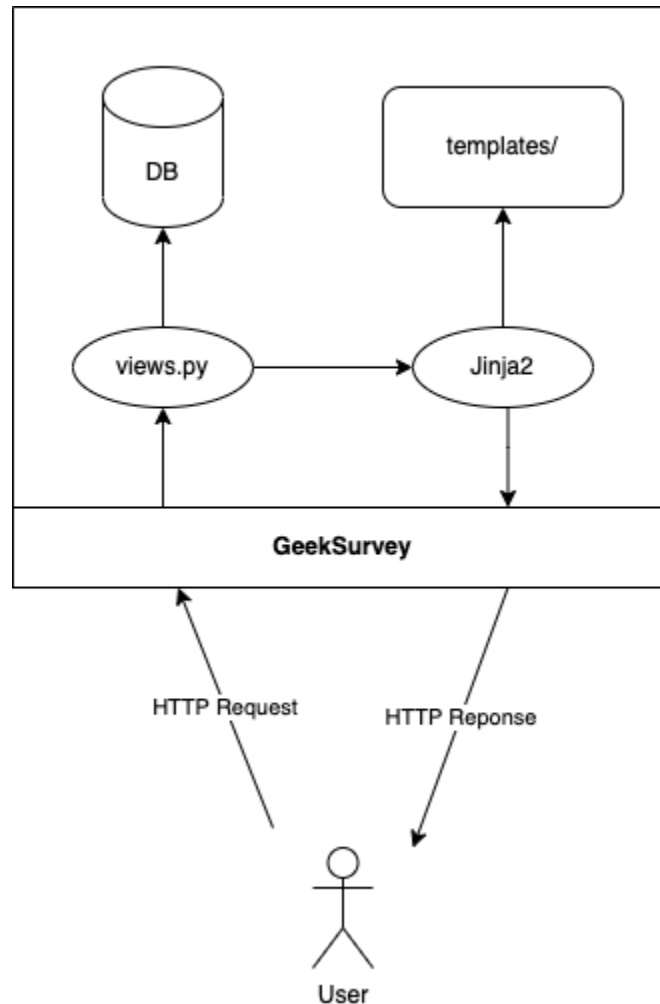
Study creation will be a simple web form. The key feature will be a link to a Google Forms survey. A major challenge in GeekSurvey's functionality is making sure GeekSurvey tracks participation data correctly for external surveys. Researchers must include a secret code at the end of their external survey, 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 Account), title, description, completion code, and external survey link.

To implement a payment system (using fake/mock payments), GeekSurvey will have to store credentials to an external GeekSurvey payment account. User accounts will 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.

### 3. Architecture Overview

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



**Figure 3.1 - 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 3.1, 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

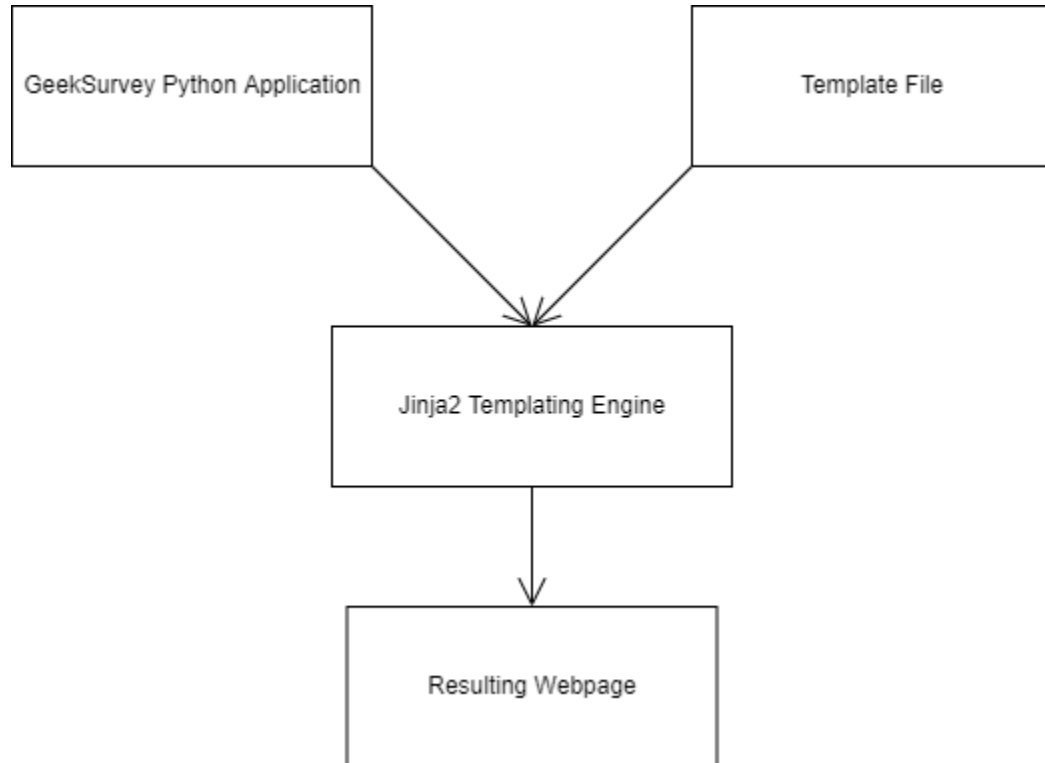
log in. If you are signed in, the same web page may show your username and some links to other parts of the web application.

## 4. Module Descriptions

GeekSurvey can be divided into three main components: the templating module, the user accounts module, and the studies module. The templating module is what allows GeekSurvey to maintain a consistent theme throughout the web application and also automatically generate pages and links. The user accounts module is what the GeekSurvey web application uses to authenticate users and allow users to update their profiles, which will be used in verifying if a user is eligible to participate in studies. The studies module is what the GeekSurvey web application uses to create studies and allows users to set participation restrictions, as well as participate in the study.



## 4.1 Templates



**Figure 4.1.1 - Overview of GeekSurvey Templating Workflow**

Much like how PHP code can effectively execute and generate HTML from within HTML code, the GeekSurvey templates follow the same idea. GeekSurvey's templating engine, Jinja2, allows for the programmatic generation of html code based on information that is pulled from the GeekSurvey web application database. By pulling information from the database, the templating engine is able to generate pages dynamically. This means that two different users can visit the same web page and be presented with different information or data that is specific to that user. This is quite convenient as otherwise, the GeekSurvey web application would only be able to serve static web pages, which are typically written exclusively in HTML and lack any kind of user-specific customization.

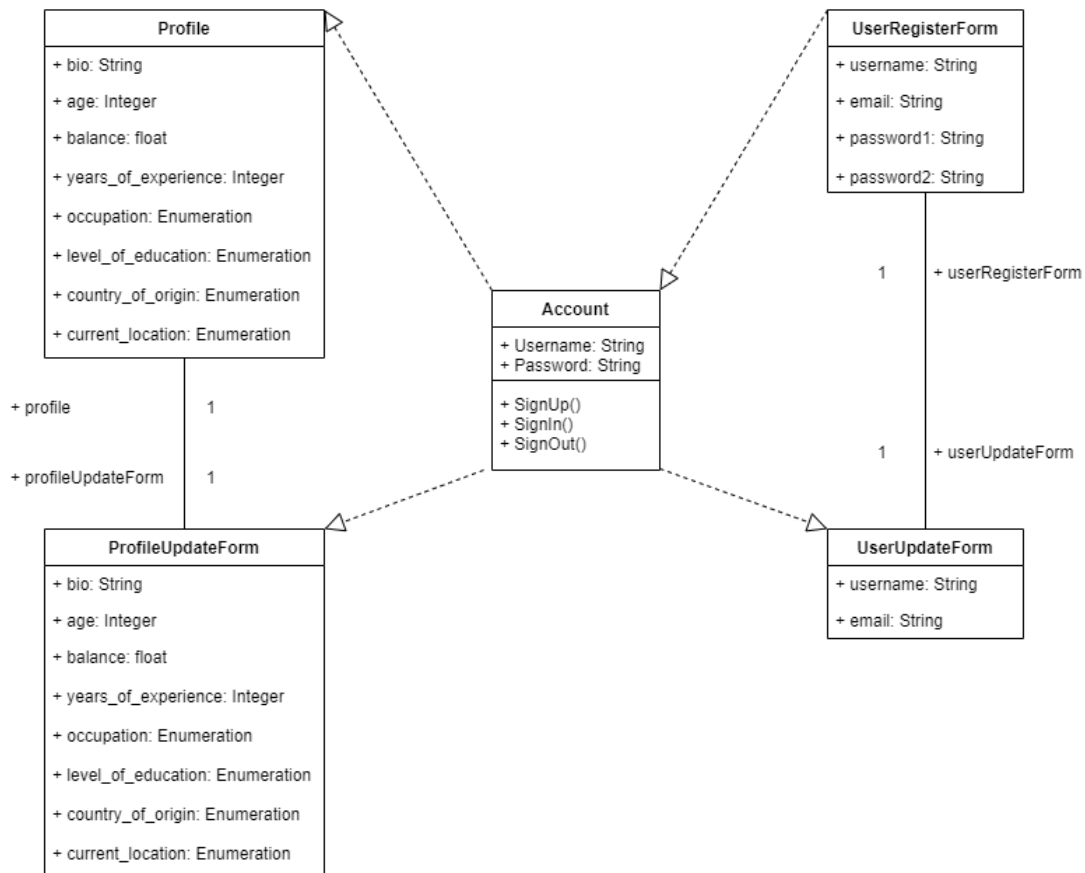
The consistent styling of the GeekSurvey web application is accomplished via the `base.html` file. The `base.html` file serves as the basis for all webpages on GeekSurvey, as it contains everything that all webpages on the site have in common. This includes the navigation bar, website header, and website footer. The middle of the `base.html` file contains blocks where the Jinja2 templating engine can generate content as required by a specific webpage.

As can be seen in Figure 4.1.1, the templating workflow works as follows. The GeekSurvey Python Application generates and gathers data from the database for use in web pages throughout the web application. The Template File, `base.html`, contains the necessary content blocks where generated HTML code will be injected into web pages. These two

components (the GeekSurvey Python Application and base.html file) are then sent to the Jinja2 Templating Engine. The Jinja2 Templating Engine then retrieves any relevant data for a webpage from the GeekSurvey Python Application's database and generates the respective HTML code, which is then injected into a new HTML file that extends the base.html file. The resulting HTML file is then served to the end user as the resulting webpage.

While this process typically requires little additional configuration, there are situations that may arise that would require further configuration. For example, the Jinja2 template for accounts defaults to a very basic login system with multiple bare pages. In order to style it for use with the GeekSurvey Web Application, the base.html file is extended and the account template is overridden.

## 4.2 User Accounts



**Figure 4.2.1 - User Account Module Interaction**

Through the user accounts module, users can sign in to GeekSurvey and update their profiles. The profiles can be edited by the users after account creation and through the profile dashboard. Information provided by users in their profile will be used by researchers when filtering for potential participants for their studies. If a user is new to GeekSurvey, they will be able to sign up and create an account. In order to sign up for an account, the user must provide an email address, username, and password. These fields can be edited at a later date. Alternatively, users can login with social accounts such as Google and GitHub.

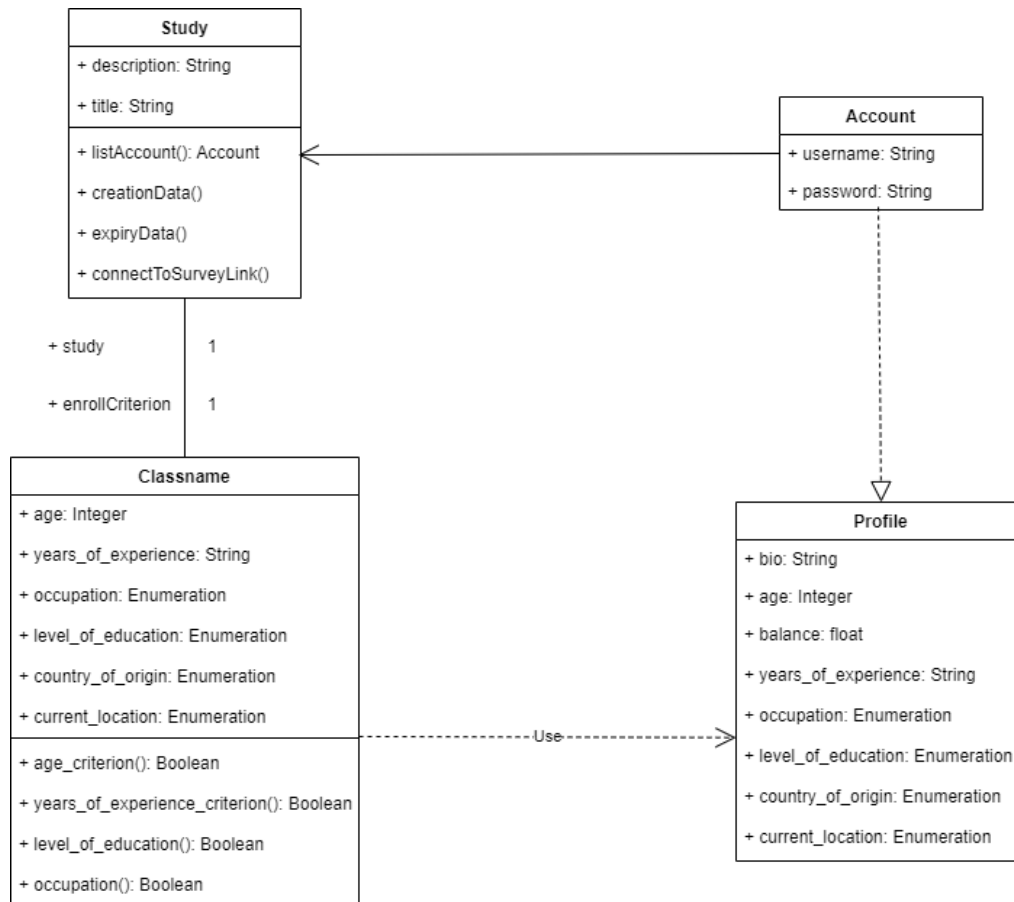
A profile in GeekSurvey is a separate database model with a one-to-one relationship to the account model. It contains a lot of basic information about the user, including biography, age, balance, user experience, occupation, and level of education. These profile fields are instrumental to one of GeekSurvey's core functionalities: enrollment criteria. Researchers often need to filter participants for a study based on certain personal information. The profile object covers basic information that researchers may want to select for. Additional fields may be added depending on client feedback.

As can be seen in Figure 4.2.1, the user account module works as follows. A new GeekSurvey user needs to create an account first. The user should be able to create a username. The user should also provide an email address and create a strong password. After finishing the registration form, the user will have an account to sign in to GeekSurvey. At the same time, the GeekSurvey database will contain a record of the new user. After creating their account, users can sign in with the username and password they have previously set up. If the user wishes to change their password, username, or email address associated with their account, then they can do so via the user update form.

Users can reset their password through the password recovery button on the sign in page. A recovery email with further instructions will be sent to the email they have registered to their account. If this operation does not work, users can contact the developers to help them retrieve their account.

Users can change almost all their profile information, except their balance. The balance can be seen but not modified and is related to the external payment system. After users modify their profile, they can submit the modified profile. Then the user's information will be updated in the database, and can be seen by other users.

## 4.3 Studies



**Figure 4.3.1 - Study Module Interaction**

The primary purpose of GeekSurvey is to provide an efficient way for software engineering researchers to conduct research and facilitate the discovery of suitable research participants. This module allows users to create and manage studies. It also allows users to be alerted to studies in which they are eligible to participate in.

As can be seen in Figure 4.3.1, the study module works as follows. When logged in, a user can create a study. The study should contain a title and a description of the research. Through the title and description, potential participants can know what the study is about, and decide if they are interested in participating. The study form also contains a creation and expiration date. The researcher can also load up a study's balance, as participants should be compensated for their contribution to the research study. The payout amount is another field that researchers may adjust. Through the payout amount, potential participants can know how much they will be compensated for taking part in the research. Researchers can also see how many participants have enrolled in his study and view the accounts of participants.

When a researcher wants to conduct a study, they need to consider who can participate in the study. Thus, the researcher can come up with enrollment criteria required of participants to partake in the study. The criteria for participation is to be based on information provided by users in their profiles. The criteria may include age, user experience, level of education, occupation, and so on. For example, if a researcher wants data scientists to participate in their study, then only those users whose occupation is data scientist will have the opportunity to participate in the research. The users with the data scientist occupation are considered as valid participants for this study, and those whose occupation is not data scientist are considered as invalid participants. Invalid participants are not allowed to enroll in the study. This operation can be accomplished by managing the database, because all the user profiles are stored in the database, and all the studies are also saved in the database. Developers of GeekSurvey can use the database to filter user information and filter out the invalid participants.

After creating these basic settings of a study, the researcher should also attach a survey link in the study for participants to participate. GeekSurvey supports external survey links and the default external survey platform is Google Forms. After creating a study, researchers can view the studies they have created. Researchers will be able to edit their studies as needed. Similar to how the user profile update process works, researchers can change study requirements and the expiration date to keep pace with the research progress.

## 5. Implementation Plan

Development for GeekSurvey follows an implementation plan which defines tasks and timelines for developers during the Spring 2022 Semester.

Figures 5.1 and 5.2 in the Appendix show the SuperGeeks Gantt Chart that is used as a schedule reference. During the first sprint, the team will be working on the home page. The home page contains a navigation menu in which users can navigate to a specific page they want to go to.

The next sprint will focus on the user account creation. During this sprint, the team will implement features that will allow users to sign up, sign in, and sign out of GeekSurvey. Users will then be able to edit and view the profile they have created upon signing up. The user profile will contain all the necessary information such as age, level of education, account balance, years of professional experience, and their occupation.

Upon the completion of the user account creation sprint, the team will then shift their focus on implementing the study creation and study discovery functionalities. This includes the ability of researchers to create a study, set a criteria for the study, and view a page of all the studies they have created on the web application. A created study will contain a title, description of the study, creation/expiration date, and a list of enrolled accounts. Studies will contain an external survey link and provide a completion code to participants upon completion and compensate them. Participants on the other hand, will be able to view studies that they are only eligible for and have not expired. Study notifications will be sent to participants that are eligible for that study.

Once the study functionalities have been implemented, the admin menu is the next functionality/feature the team plans on implementing. An admin account will be able to edit and view studies, accounts, and profiles from the admin menu.

Finally, the external payment system will be implemented. During this sprint, the team will implement an external payment system that can fund a GeekSurvey account and a study. The study will then automatically update the balance for participants that have completed the study. Users will be able to cash out their balance from the study to their account and from their account to an external bank.

## 6. Conclusion

GeekSurvey will be a web application to solve the research coordination problem identified by Dr. Gerosa and Dr. Steinmacher. 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 will offer researchers a way to conduct research and set criteria for participants to meet, as well as compensate them for completed research studies.

Currently, SuperGeeks has made amazing progress throughout the first few weeks of the Spring 2022 semester. The main functionalities are being implemented and the team is on track to provide a minimum viable product by early March. The home page, user account/profile, and study creation pages and functionalities have been implemented. SuperGeeks are meeting with the clients to get more information about user interface and design of the overall web application. The team is ecstatic about presenting the product in the Spring of 2022.



## 7. Glossary

**Jinja2:** A web template engine for Python that is similar to the Django template engine.

**PHP:** A general-purpose scripting language that is geared towards web development.



**Figure 5.2: SuperGeeks Gantt Chart Continued**

