Requirements Workshop:

1. Provide an example of five hypothetical non-functional requirements for this system. Be sure to include the specific type of requirement discussed in class, with each requirement coming from a unique category.

~Usability:

Feedback: The user should be able to receive feedback from the system. The feedback must be timely and informative for the user, such as confirming certain preferences or displaying loading screens when needed.

~Reliability:

Data Accuracy: The system should ensure that the clothing suggestion provided aligns with the users' inputted information and preferences. The system should minimize mistakes such as irrelevant suggestions.

~Performance:

Run Time: The system should be able to respond to user inputs and other queries in a fast and timely manner, reducing wait times and maximizing user satisfaction.

~Supportability:

Documentation: The system should have proper documentation, instructions, and tutorials for the user to easily understand how the system works, and what features they can use.

~Implementation/Constraints:

Integration: The system should be designed in such a way that there is easy integration access with APIs (payment APIs and online catalog APIs) to enhance the user's experience.

2. Provide an example of five hypothetical functional requirements for this system.

~User account management: The user should be able to make their account with their details such as phone, email, address, age, and weight.

~Review and rating system: The system should ask the user within a few days of using the product how the user likes the app and any suggestions for improvement.

~Account History and Order History: The system should be able to track what products the user has ordered and should be able to display it to the user at the user's discretion.

~Preferences selection: The user should be able to input their preferences when asked questions from the system to better understand what they want. Questions such as height, weight, size, color, brand, etc.

~Size and fit guide: The user should be able to review what size they want after the system provides a detailed size guide and chart description for each size.

3. Think of a specific task required to complete each of the functional requirements and non-functional requirements mentioned above (10 total). Estimate the amount of effort needed to complete this task using function points (i.e., using the values here). Briefly explain your answer.

1. User Account Management: Task: Implement user registration form. This task involves designing and coding a form where users can input their details like phone, email, address, age, weight, and height. We estimate this task to use up 5 function points.
2. Review and Rating System: Task: Develop a feedback prompt and storage mechanism. This task involves creating a feedback prompt that appears after a few days of using the app, collecting user suggestions, and storing them for analysis. Estimated 3 function points.
3. Account History and Order History:Task: Design order tracking functionality. This task involves creating a mechanism to track user orders and display them to users when requested. Estimated 8 function points.
4. Preferences Selection:Task: Develop preference input forms and processing logic. This task involves creating forms where users can input their preferences like height, weight, size, color, brand, etc., and implementing logic to process and store these preferences. Estimated 3 function points.
5. Size and Fit Guide: Task: Create a detailed size guide feature. This task involves designing and implementing a feature that provides users with detailed size guides and chart descriptions for each size. Estimated 2 function points.
6. Usability:Task: Implement feedback messages and loading screens. This task involves designing and coding feedback messages to confirm user preferences and implementing loading screens where necessary. Estimated 2 function points.
7. Reliability : Task: Ensure data validation and accuracy in clothing suggestions. This task involves implementing validation checks to ensure that clothing suggestions align with

user preferences and minimizing errors such as irrelevant suggestions. Estimated 5 function points.

8. Performance: Task: Optimize system response time. This task involves analyzing and optimizing system performance to reduce response times for user inputs and queries. Estimated 8 function points.

9. Supportability: Task: Develop user documentation and tutorials. This task involves writing and organizing user manuals, instructions, and tutorials to help users understand the system's features and usage. Estimated 3 function points.

10. Implementation/Constraints: Task: Integrate with payment and online catalog APIs.This task involves integrating the system with external APIs for payment processing and accessing online catalog data to enhance the user experience. Estimated 5 function points.

4. Write three user stories from the perspective of at least two different actors. Provide the acceptance criteria for these stories.

Story 1: From the perspective of a customer: As a customer, I would like to be able to get suggestions in a short period and would like to easily apply my filters. The acceptance criteria for this story would be that the suggestions would appear within 4-5 seconds and filters must be easy to find and apply and be able to load the suggestion page.

Story 2: From the perspective of a customer: I want to be able to quickly check out the product I have chosen which includes a simple checkout process with confirmation. The acceptance criteria for this would be that the checkout process takes at most 4 steps, and after submitting payment, the system displays an order summary of all the information so the user has confirmation.

Story 3: From the perspective of an administrator: As an administrator, I want to be able to review how many orders have been placed from my service a day and how many per customer. The acceptance criteria for this would be for the system to be able to provide a detailed summary of orders for each day and the total price.

5. Provide two examples of risk that could impact this project. Explain how you would mitigate these risks if you were implementing your project as a software system.

~Security risk and data breach: As a system that will have access to personal data of the users such as address, card information, name, phone, etc. security breach is a huge risk. The mitigate this risk we could implement security measures such as using secure coding practices, and encryption.

~System Downtime: Another risk our system may face is large amounts of traffic at one time. Which can cause glitches, lags, and customer dissatisfaction. The way to mitigate this risk is to choose a reliable hosting service and regular testing and maintenance which will ensure the the system is running smoothly during high traffic times.

6. Describe which process your team would use for requirements elicitation from clients or customers, and explain why.

Our team would use interviews, focus groups, questionnaires, and surveys. The interviews and focus groups would help us understand the needs of the customers, as well as any questions or concerns they may have. Our team would use questionnaires and surveys to gather any specific data on user preferences in terms of qualitative data such as how many suggestions they would like to receive when entering their preferences.

Requirements Analysis:

1. **Use Case 1:** Shopping for a cheap blue top

   Primary Actor: Registered User

Stakeholders & Interests: The registered user wants to be able to buy a cheap blue top, and the stakeholder wants to be able to provide an easy seamless shopping experience for the user.

Preconditions: The user must have already created an account and put in the information requested such as email and address. Must also be a registered user with reliable internet access.

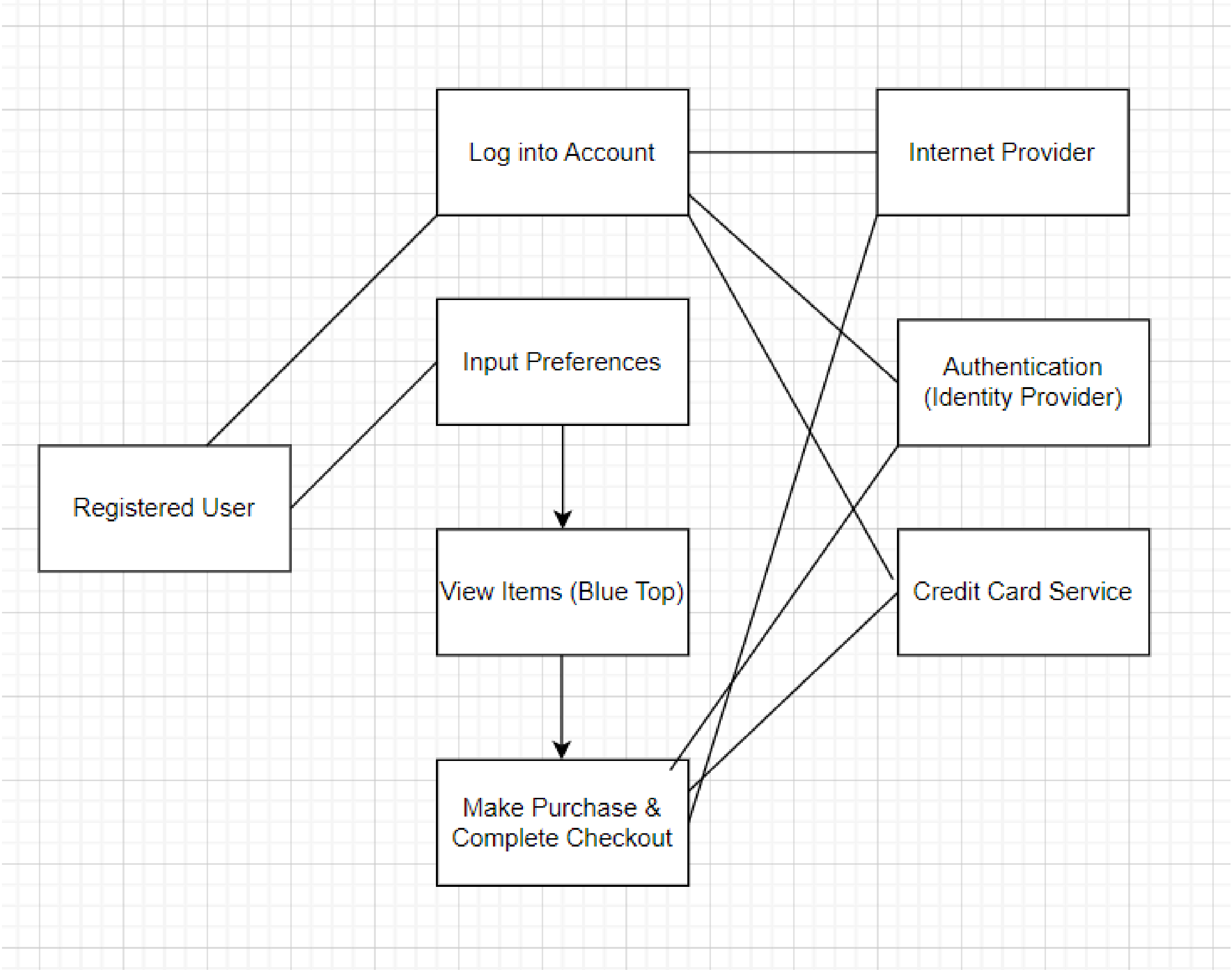
Postconditions: The user can receive a link to purchase the blue top and later the application will be able to update their preferences and save it within their profile.

Main Success Scenario: The user will put in preferences for the clothing article they are looking for such as preference of color, type of clothing, and price range. [S1]. The app will return clothes that fit preferences based on previous choices as well. User will choose a top to their liking [S2]. The user chooses an option they like and buys it on the site through the app [S3]. Users will be taken to the site selling the top and will be able to order the top to their address and AI will take note of the user choice when providing future recommendations [S4]. The user buys the top of their choice [S5].

Alternative Flows: If the system is not able to determine the user's location then the user will be prompted with a message box asking the user to manually enter their zip code within a mile radius for locations of stores.

Special Requirements: Personal data such as card information, address, and user preferences must be handled securely to comply with security requirements.

Use Case Diagram:



2. **Use Case 2:** User inputting preferences

   Primary Actor: Registered user

Stakeholders & Interests: The user can easily follow directions to input preferences, and the stakeholder wants to be able to provide an easy seamless shopping experience for the user.

Preconditions: The user must have already created an account and put in the information requested such as email and address. Must also be a registered user with reliable internet access.

Postconditions: The user can input their preferences and is now ready to receive links and ready to start shopping.

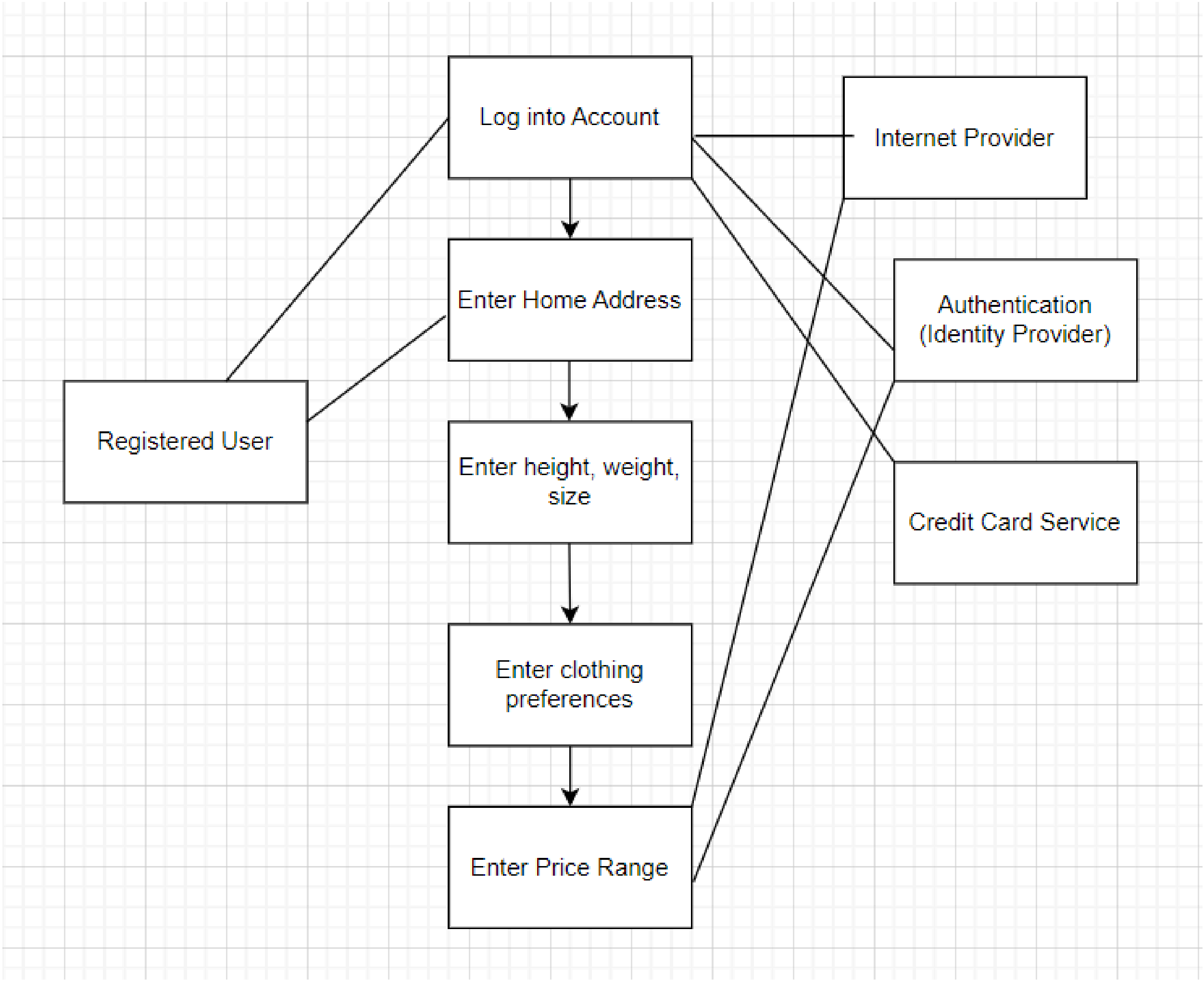
Main Success Scenario: The user logs in [S1]. The user is now prompted to input their current home address [S2]. The user is prompted to input their height and weight along with their size [S3]. The user is prompted to input the type of clothing they want and what style of clothing [S4]. The user can input their price range for clothing [S5].

Alternative Flows: If the system is not able to determine the user's location then the user will be prompted with a message box asking the user to manually enter their zip code within a mile radius for locations of stores.

Special Requirements: Personal data such as card information, address, and user preferences must be handled securely to comply with security requirements.

:

Use Case Diagram:



3. **Use Case 3:** Shop for a specific type of clothing (Jeans)

   Primary Actor: Registered User

   Stakeholders & Interests: At the end, the user can successfully input that they want to search for jeans and is provided with numerous options for jeans around them, and the stakeholder wants to be able to provide an easy seamless shopping experience for the user.

   Preconditions: The user must have already created an account and put in the information requested such as email and address. Must also be a registered user with reliable internet access.

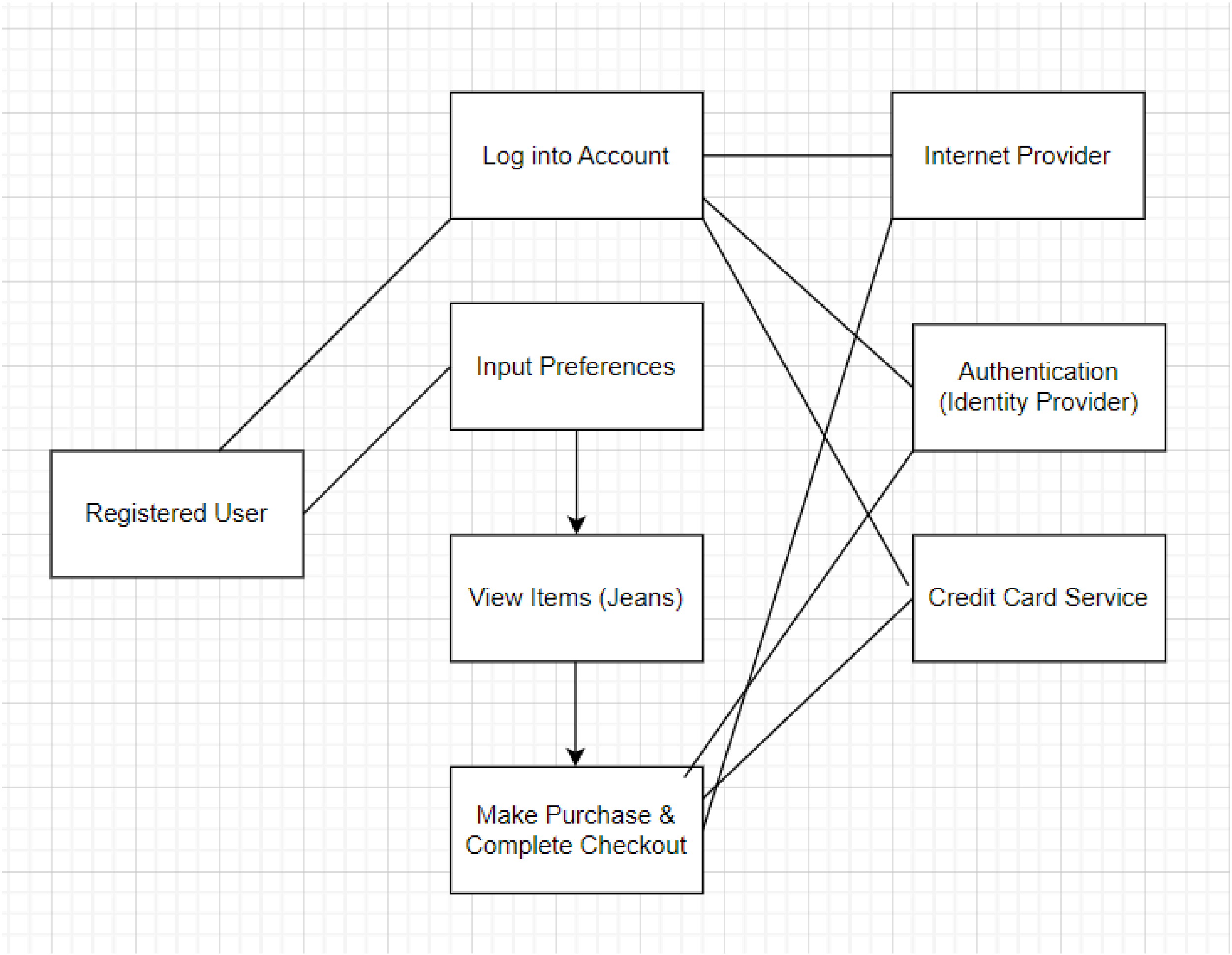
   Postconditions: The user can receive recommendations for jeans and can click on the links to receive those specific recommendations and start purchasing.

Main Success Scenario: The user logs into their account [S1]. The user is now prompted to input their current home address [S2]. The user is prompted to input their height and weight along with their size [S3]. The user is prompted to input the type of clothing they want and what style of clothing. In this case the user inputs jeans for clothing type[S4]. The user can input their price range for clothing [S5]. The user receives recommendations for jeans in stores near them [S6]. The user can click on those links and can start to purchase jeans [S7].

Alternative Flows: If the system is not able to determine the user's location then the user will be prompted with a message box asking the user to manually enter their zip code within a mile radius for locations of stores.

Special Requirements: Personal data such as card information, address, and user preferences must be handled securely to comply with security requirements.

Use Case Diagram:



4. **Use Case 4:** The user wants to change their delivery address

Primary Actor: Registered User

Stakeholders & Interests: The user can easily change their delivery address without taking time away from shopping, and the stakeholder wants to be able to provide an easy seamless shopping experience for the user.

Preconditions: The user must have already created an account and put in the information requested such as email and address. Must also be a registered user with reliable internet access. The user has also imputed their delivery address at a previous time before.

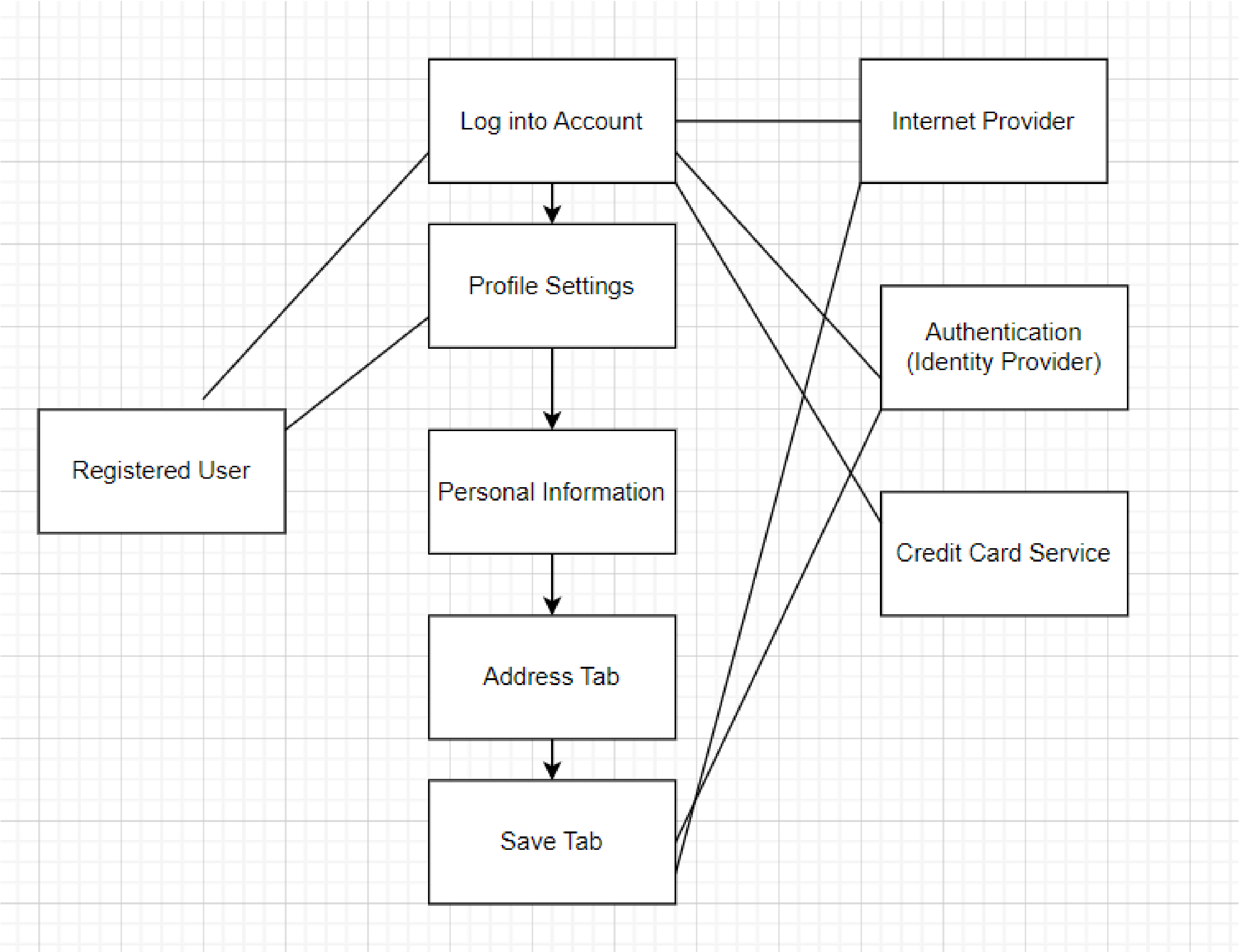
Postconditions: The user has successfully changed their delivery address and can now start shopping.

Main Success Scenario: The user logs into their account [S1]. The user goes to profile settings [S2]. The user then clicks on the personal information tab [S3]. The user clicks on the address tab [S4]. The user is now able to input a new address and can save this address for future deliveries [S5].

Alternative Flows: If the system is not able to determine the user's location then the user will be prompted with a message box asking the user to manually enter their zip code within a mile radius for locations of stores.

Special Requirements: Personal data such as card information, address, and user preferences must be handled securely to comply with security requirements.

Use Case Diagram:



5. **Use Case 5:** The user wants to change their basic profile information (height, weight, and size.)

Primary Actor: Registered User

Stakeholders & Interests: The user is successfully able to change their basic profile information including height, weight, and size, and the stakeholder wants to be able to provide an easy seamless shopping experience for the user.

Preconditions: The user must have already created an account and put in the information requested such as email and address. Must also be a registered user with reliable internet access. The user must already have imputed their basic profile information which includes their height, size, and weight.

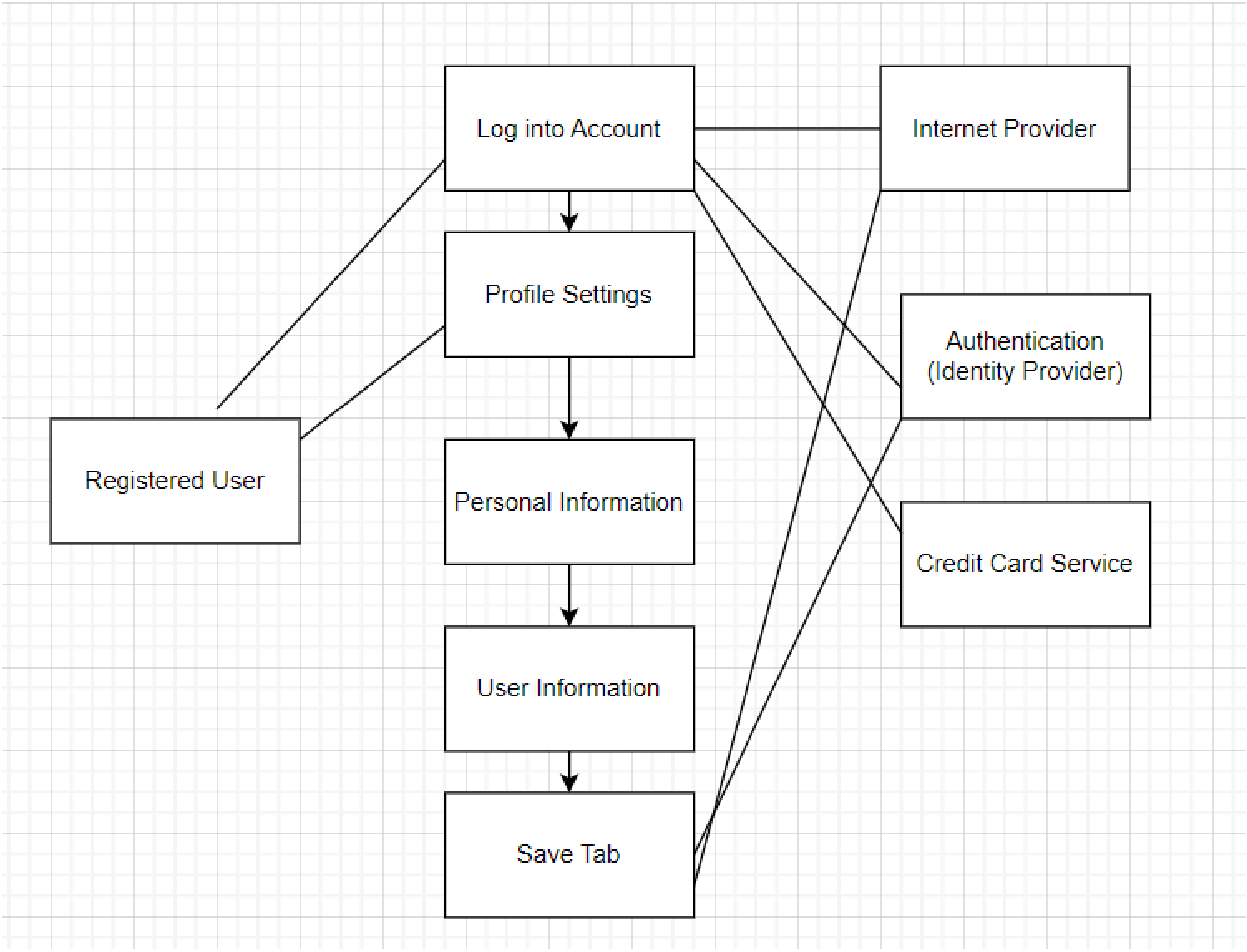
Postconditions: The user has successfully changed their basic profile information and is ready to continue shopping.

Main Success Scenario: The user logs into their account [S1]. The user goes to profile settings [S2]. The user then clicks on the personal information tab [S3]. Within the personal information tab, the user clicks on the user information tab [S4]. The user is then able to change their height, weight, and size within this tab [S5]. The user exits this tab and can now continue shopping [S6].

Alternative Flows: If the system is not able to determine the user's location then the user will be prompted with a message box asking the user to manually enter their zip code within a mile radius for locations of stores.

Special Requirements: Personal data such as card information, address, and user preferences must be handled securely to comply with security requirements.

Use Case Diagram:



Process Deliverable: Adaptive Software Development

For our proposed application, the agile methodology we plan to use is Adaptive Software Development (ASD). We chose this approach because agile development focuses on iterative development when requirements and project specifications are constantly changing and evolving. Our application is a recommendation system for clothes based on our users' preferences and filters. Based on their preferences the application will suggest specific clothes, brands, or types of clothing to the user. In regards to our application, agile methodologies will allow us to quickly develop an AI/ML recommendation system and user interface. It will also allow for the application to refine itself and adjust on its own based on the interactions with the user and the user's feedback. Because of Adaptive Software Development, our application for example will allow a user to input their preferences and weight, size, and height and will continuously adapt and change the preferences it provides. This is why user feedback and interaction are vital for our application to perform at its highest level and reach its potential. Additionally, continuous testing will ensure that the application is working well and that there are no glitches/bugs. Another advantage of ASD that pertains to our application is sprint planning this allows for us to divide the development process into shorter periods that are usually 2-4 weeks long as this allows for shorter and quicker iterations of development time. Although many other software development techniques could be used for our application, the best one seems to be adaptive software development because of the many advantages it provides when creating a user-interactive interface and application.