**Research Labs Inventory**

Jonathan Moore

**web application subsystem final report**

REVISION – Final

5 December 2024

Subsystem final report

for

Research Lab Inventory

Prepared by:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Author Date

Approved by:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Project Leader Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

John Lusher, P.E. Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

T/A Date

**Change Record**

| **Rev.** | **Date** | **Originator** | **Approvals** | **Description** |
| --- | --- | --- | --- | --- |
| **1.0** | 12/5/2024 | Jonathan Moore |  | Final Release |

**Table of Contents**

[**Table of Contents III**](#_heading=h.30j0zll)

[**List of Tables IV**](#_heading=h.1fob9te)

[**No table of figures entries found. IV**](#_heading=h.49x2ik5)

[**List of Figures V**](#_heading=h.3znysh7)

[**1.**](#_heading=h.2p2csry) **Introduction 6**

**2**[**.**](#_heading=h.2p2csry) **Development Environment and Tools 7**

**3**[**.**](#_heading=h.23ckvvd) **Data Preparation 8**

3[.1.](#_heading=h.ihv636) Data Setup 8

3[.2.](#_heading=h.32hioqz) Database Configuration 8

3[.3.](#_heading=h.1hmsyys) Data Validation 8

**4**[**.**](#_heading=h.41mghml) **Application Design 9**

4[.1.](#_heading=h.2grqrue) Logo 9

4[.2.](#_heading=h.vx1227) Fonts 9

4[.3.](#_heading=h.vx1227) Intuitive Design 10

**5**[**.**](#_heading=h.3tbugp1) **Features 11**

5[.1.](#_heading=h.2grqrue) Registration and Sign In 11

5[.2. Search and Filter](#_heading=h.vx1227) 13

5[.3. Checkout](#_heading=h.vx1227) 14

5.4. Account Settings 15

5.5. Management 17

**6. Validation and Testing 20**

**7. Challenges 21**

**8. Integration Plans 22**

**9. Future Improvements 23**

**10. Conclusion 24**

**11. References 25**

**List of Tables**

None.

**List of Figures**

[**Figure 1. LabRat Logo**](#_heading=h.1mrcu09) 9

[**Figure 2. White LabRat Logo with Title**](#_heading=h.46r0co2) **9**

[**Figure 3. Home Page**](#_heading=h.1mrcu09) 11

[**Figure 4. User Dropdown**](#_heading=h.46r0co2) **12**

[**Figure 5. Registration Page**](#_heading=h.1mrcu09) 12

[**Figure 6. Sign In Page**](#_heading=h.1mrcu09) **13**

[**Figure 7. Search Feature**](#_heading=h.1mrcu09) 13

[**Figure 8. Cart Icon**](#_heading=h.1mrcu09) **14**

[**Figure 9. Cart**](#_heading=h.1mrcu09) 14

[**Figure 10. Cart Agreement**](#_heading=h.1mrcu09) **15**

[**Figure 11. Account Page**](#_heading=h.1mrcu09) 16

[**Figure 12. Edit Account Modal**](#_heading=h.1mrcu09) **16**

[**Figure 13. Color Modal**](#_heading=h.1mrcu09) 17

[**Figure 14. Color After Selection**](#_heading=h.1mrcu09) **17**

[**Figure 15. Management**](#_heading=h.1mrcu09) 17

[**Figure 16. Users Table**](#_heading=h.1mrcu09) **18**

[**Figure 17. Items Table**](#_heading=h.1mrcu09) 18

[**Figure 18. New Item Modal**](#_heading=h.1mrcu09) **19**

[**Figure 19. Orders Table**](#_heading=h.1mrcu09) 19

[**Figure 20. Playwright Terminal Output**](#_heading=h.1mrcu09) **20**

[**Figure 21. HTML Output**](#_heading=h.1mrcu09) 20

**Web Application Subsystem Report**

**1.Introduction**

For this project, it was necessary to create a website that any user at the laboratory could access at any time to manage their items in the lab. This website was designed to be a tool that people can interact with at a simple level without having to know how to use a database. Management of users (such as students, professors, administrators, etc.) is simple and intuitive and allows for many different permission levels for the different types of people that will be interacting with our project. After users check out items, there are certain types of items that need to be returned to the laboratory, and this website tracks that information and will send email reminders to users once full database integration takes place.

**2. Development Environment and Tools**

The web application was developed without a web framework. This was a deliberate choice by myself so that development could move as quickly as possible since there were a lot of goals set for this subsystem at the beginning of the semester. I used a combination of HTML, CSS, and Javascript to manage the frontend of the website, and Node.js, SQLite3, Python, and additional Javascript to handle all server-side responsibilities. For all validation, I created regression tests and input validation tests using Playwright. With all of these tools, the web application was successfully completed and is ready for integration.

**3. Data Preparation**

**3.1 Data Setup**

Using SQLite3, I have a data set of placeholder information that gets re-seeded each time the website boots up on a local server. There were many times I’ve had to completely wipe the database for testing reasons during development, so I have written a python script to run each time the server notices an empty database at run time. Using this, I was able to move through development very smoothly. This will not appear in the final product, as integration would eliminate the need for a local database, but it is important to mention it since there would be no displayable functionality without it.

**3.2 Database Configuration**

The database structure is kept very similar to the other subsystem that was made for this purpose, with minor alterations to fit the needs of the website; all of these changes are miniscule and will take minutes to change once real integration begins. Speaking of which, there is no need to worry about the website not connecting to the database correctly next semester, as we have all stuck with using uniform SQL formatting so that all we will need to do is “unplug” our local databases in favor of the new, live database.

**3.3 Data Validation**

All inputs that the user gives through the website are tested and validated depending on what is entered into each input field of the application. If certain fields are required, then the website will alert the user of incomplete information and block it from reaching the database. Emails are handled differently than other pieces of information on the application. Since emails should be unique to the user, only one account can be registered per email. On top of this, since sending emails is planned with database integration, each email must be valid upon registration. There is a simple validation check for email formatting, so invalid email types are not allowed during registration. This is just one example of protection the website provides to the database and its users.

**4. Application Design**

**4.1 Logo**

Below, Figures 1 and 2 are logos that I designed at the beginning of the semester. These were designed using Adobe Illustrator, so we own the right to use these images. The name is a nod to the popular idea of testing rats in laboratories, with the rat image resembling an Erlenmeyer flask. Both of which could be found in labs, so the idea stuck.

****

**Figure 1: LabRat Logo**

****

**Figure 2: White LabRat Logo with Title**

**4.2 Fonts**

All fonts used across the app are publicly available, and we have the licenses to use them. The fonts are also shared with the mobile application for parity between systems.

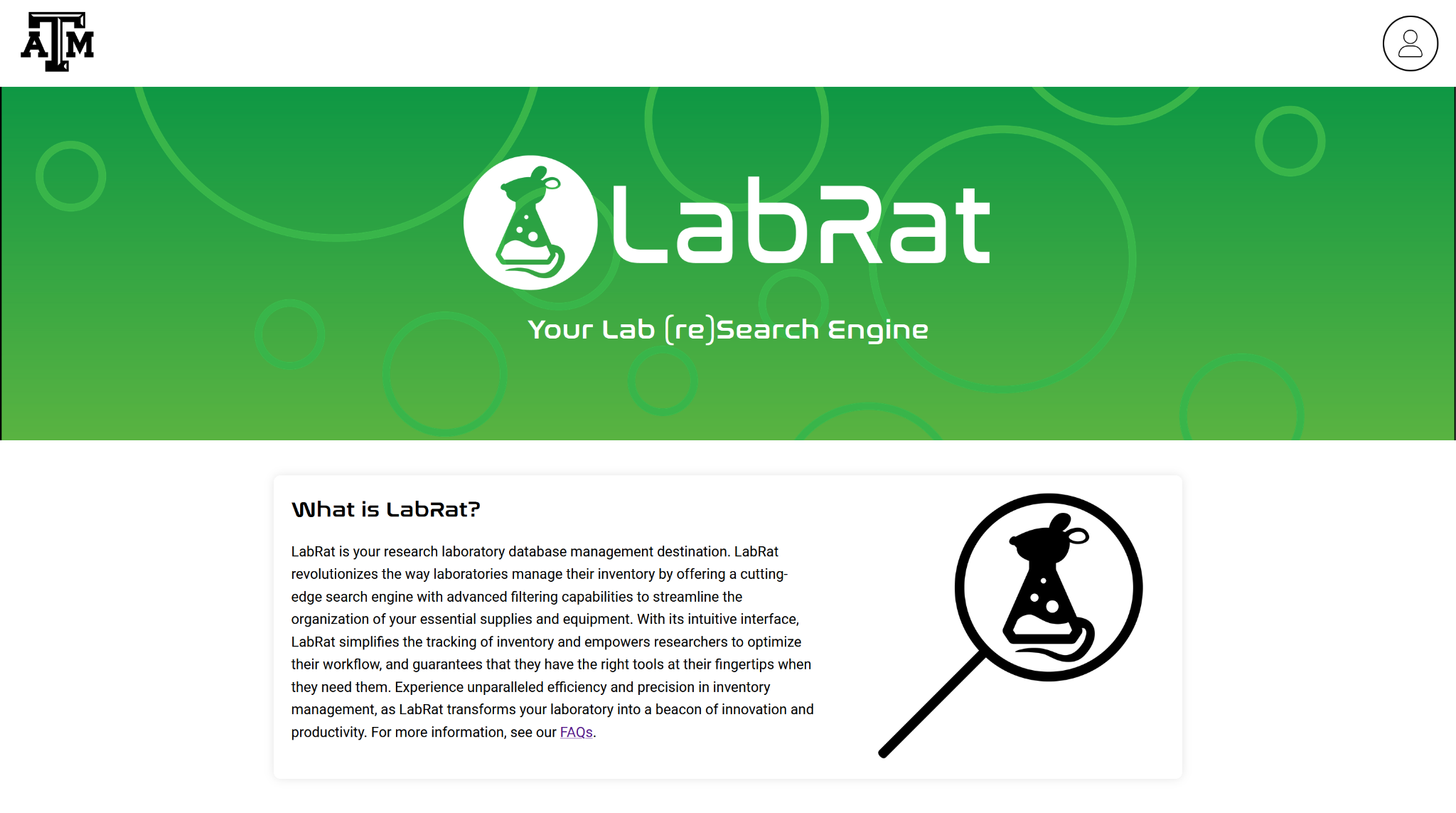
**4.3 Intuitive Design**

As for the layout and structure of the website’s appearance, it is designed to be as user-friendly as possible. With experience in creating accessibility-focused websites, I set a goal to achieve a simple, clean look that can guide users through its pages without having pages of text to read. Using icons that convey functionality (gear icons for settings, person icons for account, shopping cart icons for the cart, etc.) help declutter all of the pages while also providing meaning to the user. Tab ordering for each page was a focal point; people who may not be able to use a mouse (because of restrictions based on physical ability or unavailability of a mouse) are still able to tab through each page, type what they need to, and hit enter to send information. To summarize, intuitive navigation and accessibility was a big focus of the project, and has led to quick understanding from test users.

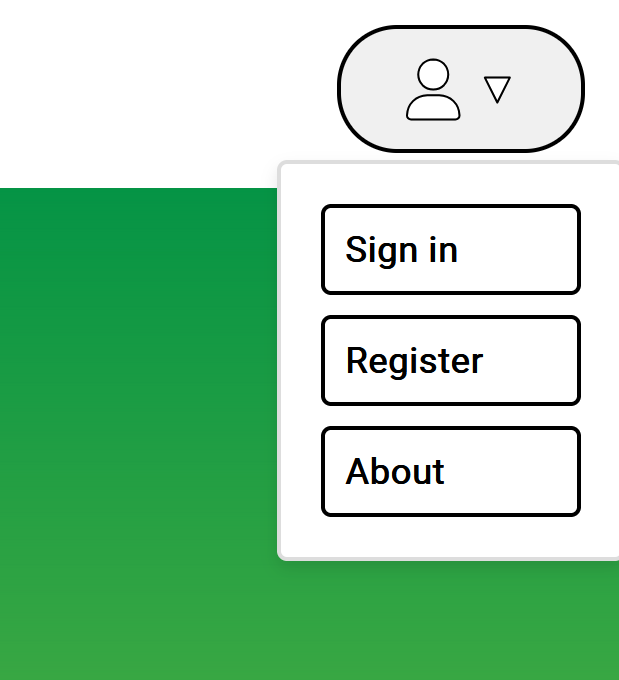
**5. Features**

**5.1 Registration and Sign In**

All users land on the homepage without a login. From here (Figure 3), users can read a small passage clueing them on what our project is about. From there, users can access a Frequently-Asked-Questions (FAQ) page that can offer more guidance.

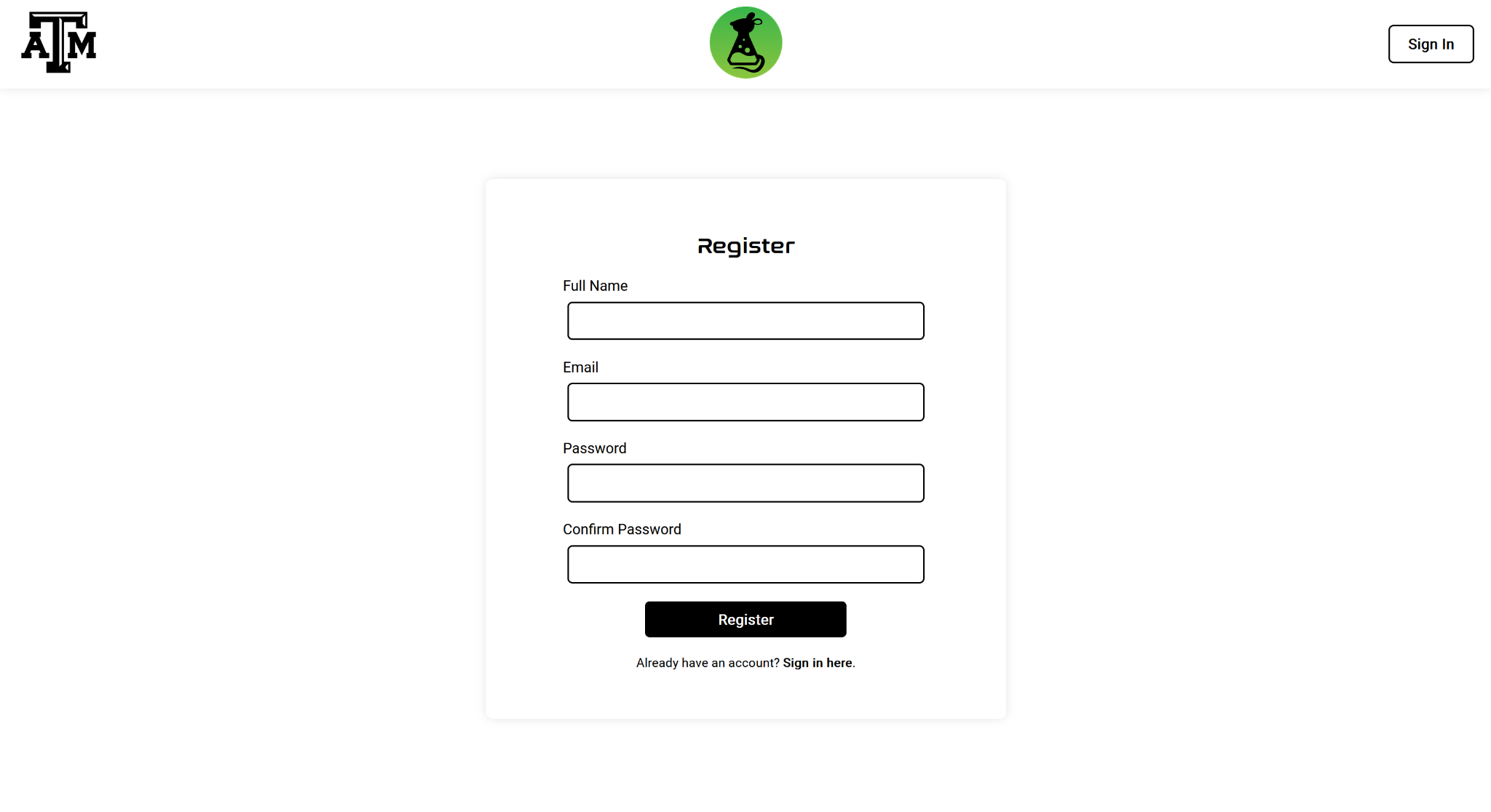


**Figure 3: Home Page**To register for an account, users click the person icon on the right which expands for more options.

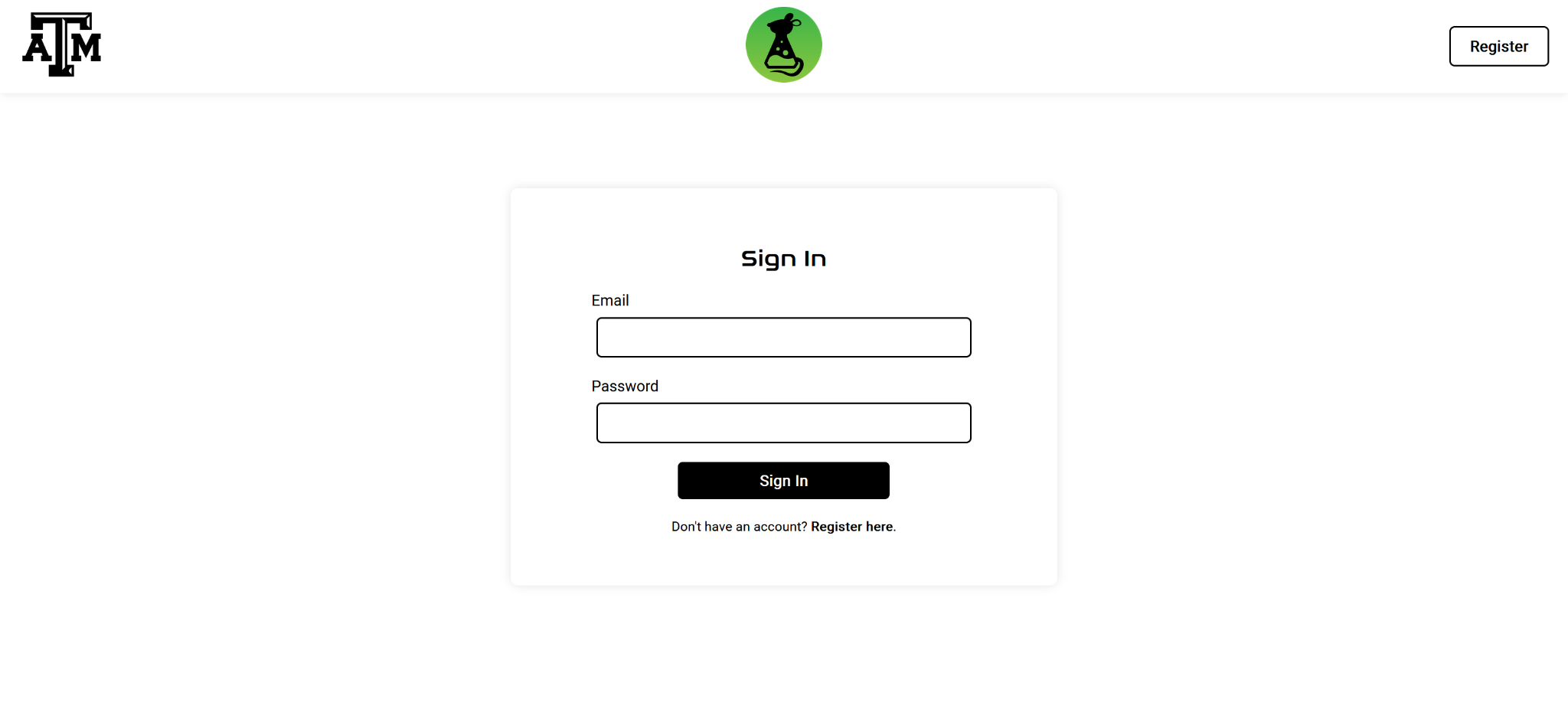


**Figure 4: User Dropdown**

From here, users have the option to Sign in, Register, and learn more about LabRat. Below are the registration and sign-in pages. Each function exactly how you’d expect. Users must register an account before being able to login, and the website alerts the user if emails or passwords are incorrect upon sign-in.



**Figure 5: Registration Page**

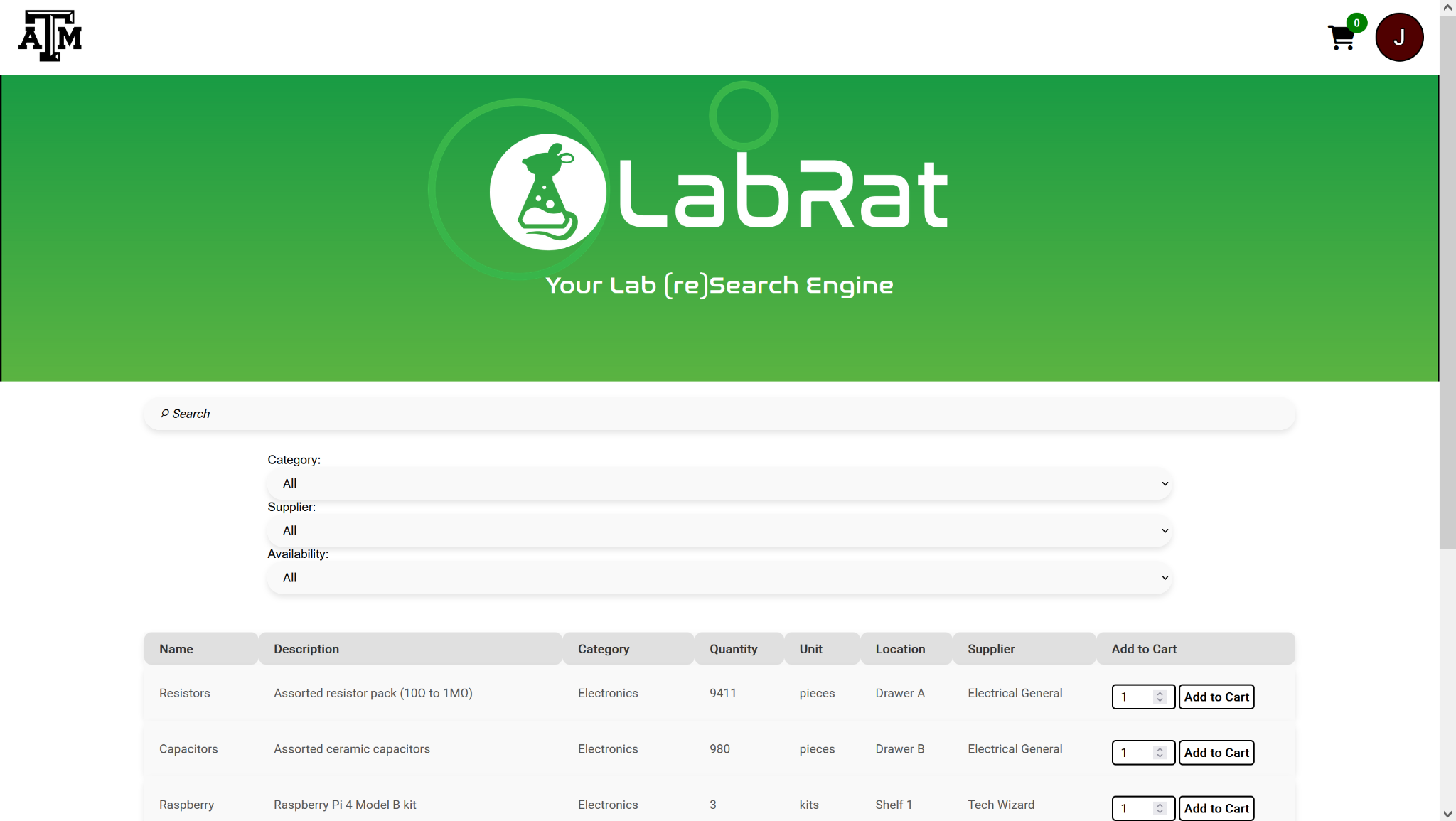
****

**Figure 6: Sign In Page**

After signing in, users are automatically routed to the home page, which is now repopulated with new information and features. Every new account starts off as a student account, so if users need permission level changes, they must consult a currently active Administrator. All of this information is available in the FAQ page.

**5.2 Search and Filter**

After signing in, the home page appears like it does in Figure 7. It consists of a search bar, filters for category, supplier, and availability, and a table filled with items from the lab.



**Figure 7: Search Feature**

On each entry in the table, users are able to select how many of a chosen item they want to checkout and add them to their cart. Items that are out of stock are greyed out and do not allow people to use them.

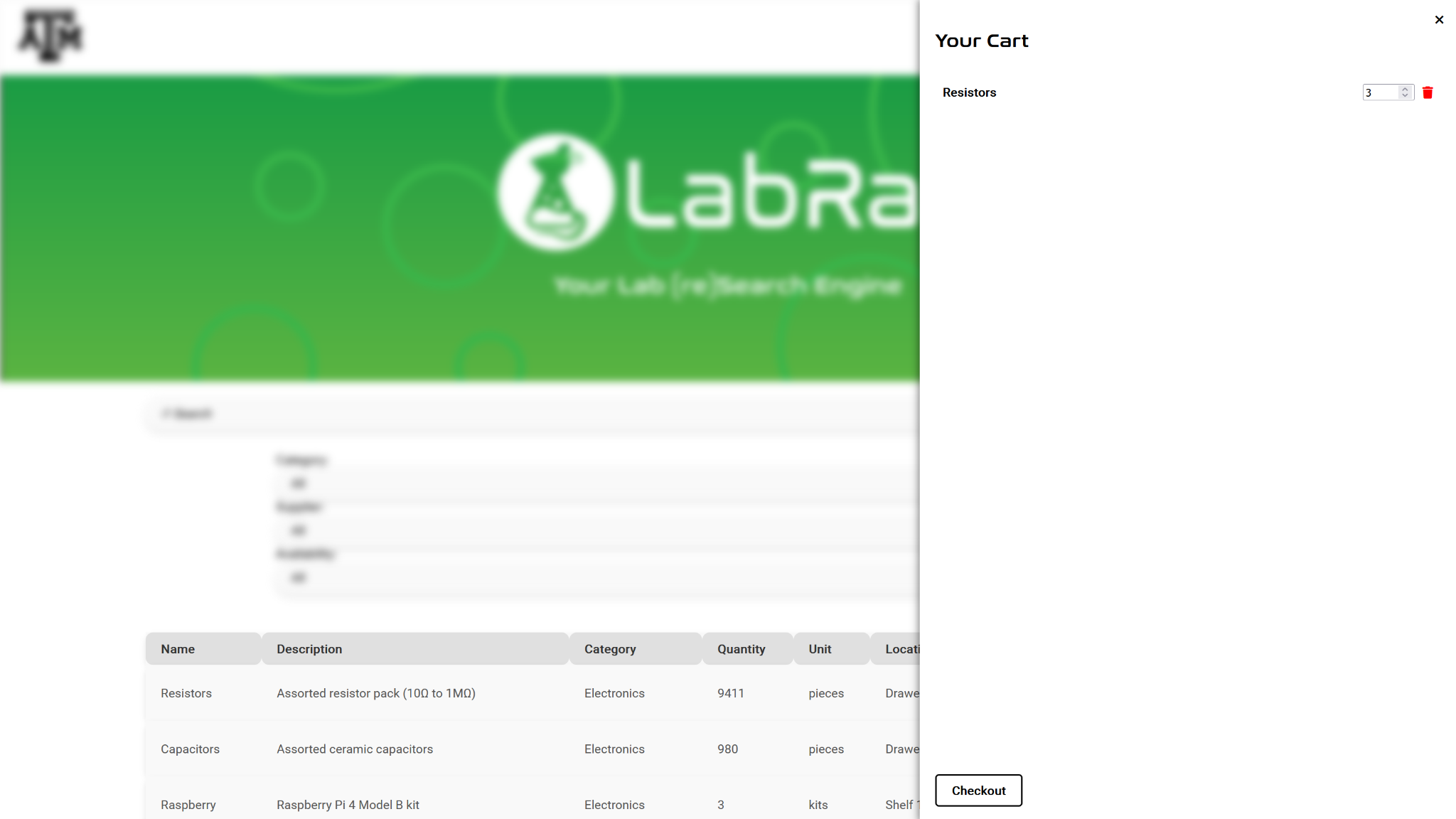
**5.3 Checkout**

Once items are added to the user’s cart, the counter bubble on the shopping cart icon next to the user’s initial increases with the number of selected items.



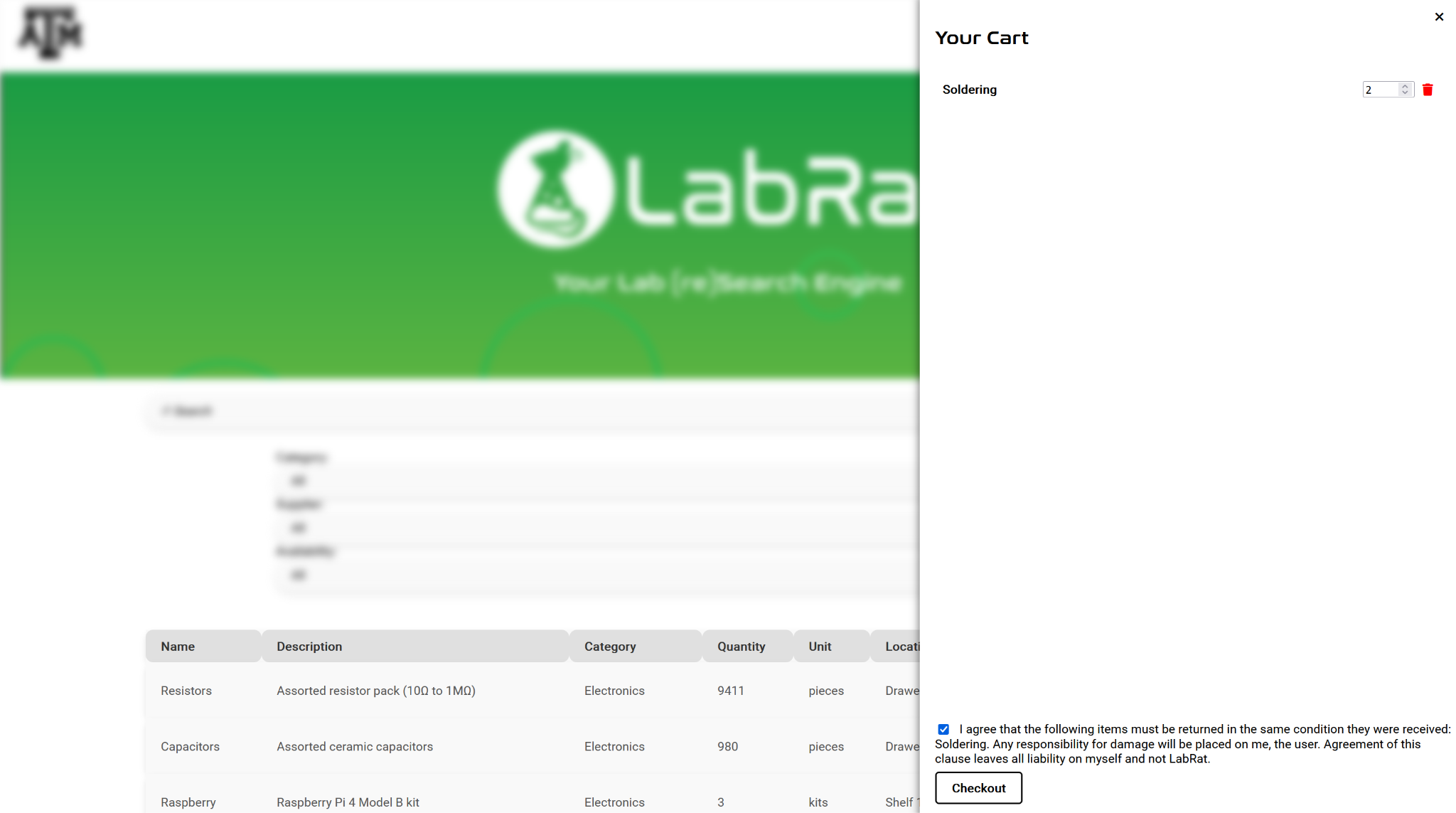
**Figure 8: Cart Icon**

After this, the user can click the cart to view its contents. They are listed in bundled quantities, and users can change the quantity of an item or remove it entirely from the cart as well.



**Figure 9: Cart**

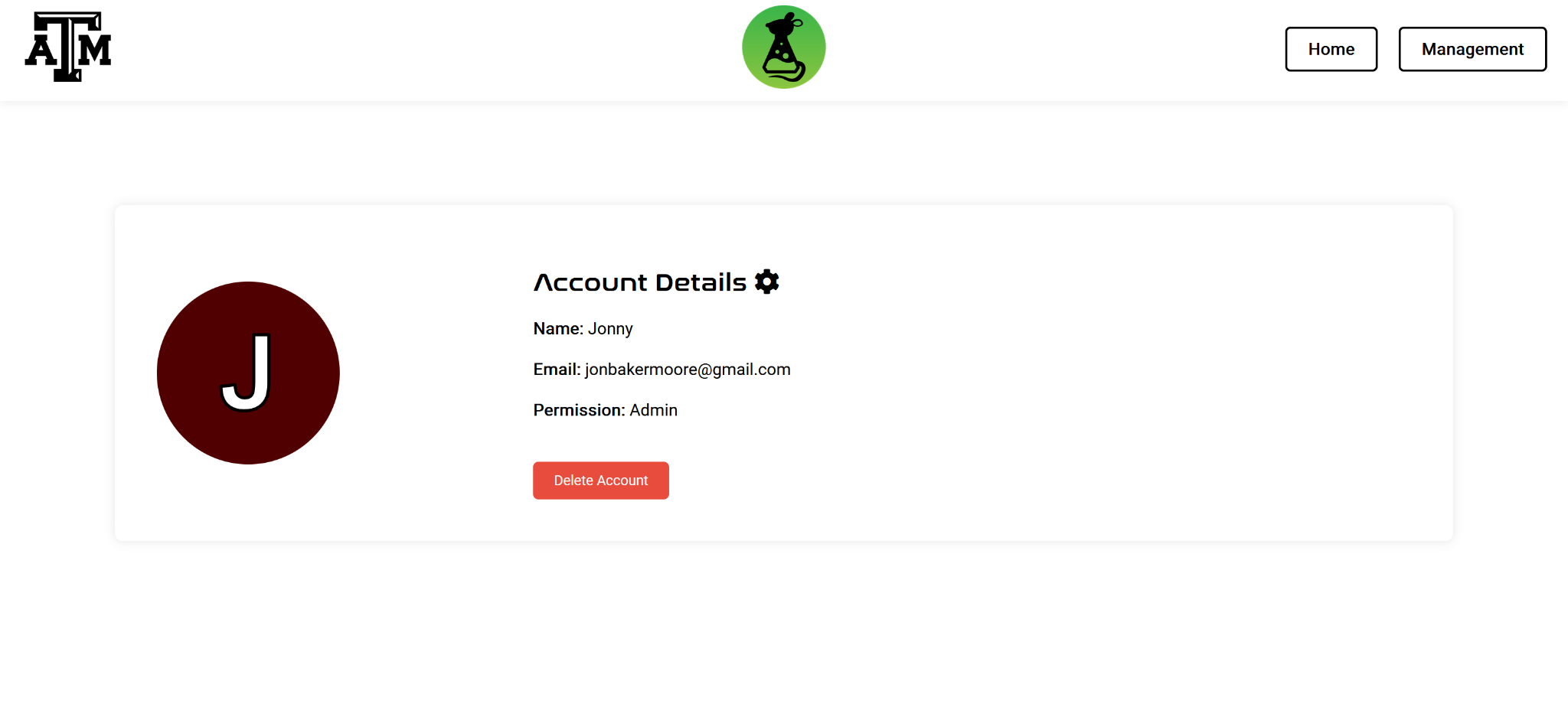
After clicking checkout, the stock decreases in the database, and the order is added to the orders table for management to observe. However, if an item belongs to the lab and needs to be returned eventually (soldering irons, etc.), then an agreement checkbox appears for the user to sign. This agreement holds users liable for damage of lab property if returned in poor condition or missing. After clicking the agreement, then the checkout button appears for the user. This time, the checkout operates differently. Each time a user checks out a returnable item, it is flagged and added to a list of items that user needs to eventually return. All contents of the list are observable from management’s point-of-view, and once database integration is complete, automatic emails can be sent to users reminding them of their agreement and responsibilities.



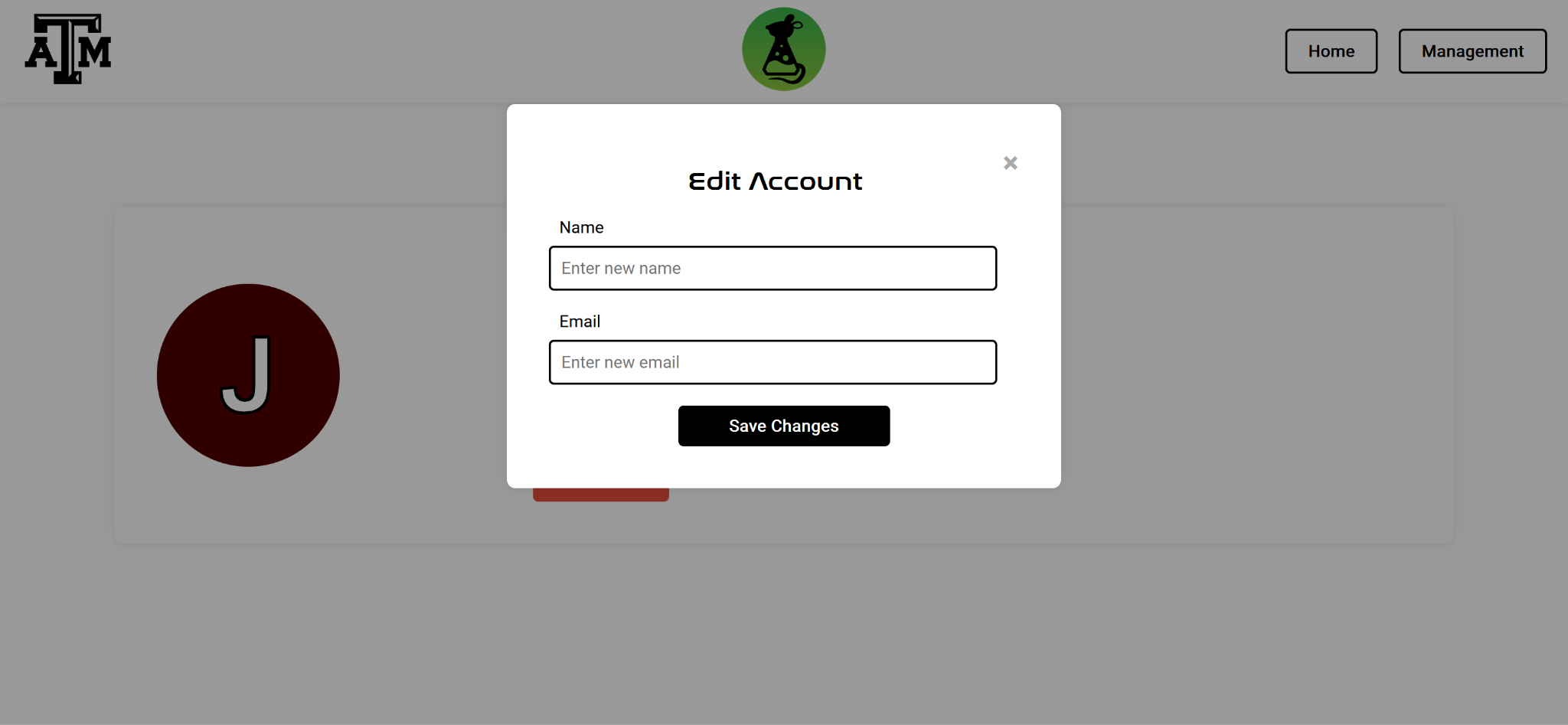
**Figure 10: Cart Agreement**

**5.4 Account Settings**

Every user has the ability to change their email, name, and profile color from the account page. Here, you can see information associated with your account. To change user information, users can click the gear icon to change them. The delete account button also works, but it warns the user and asks for confirmation.

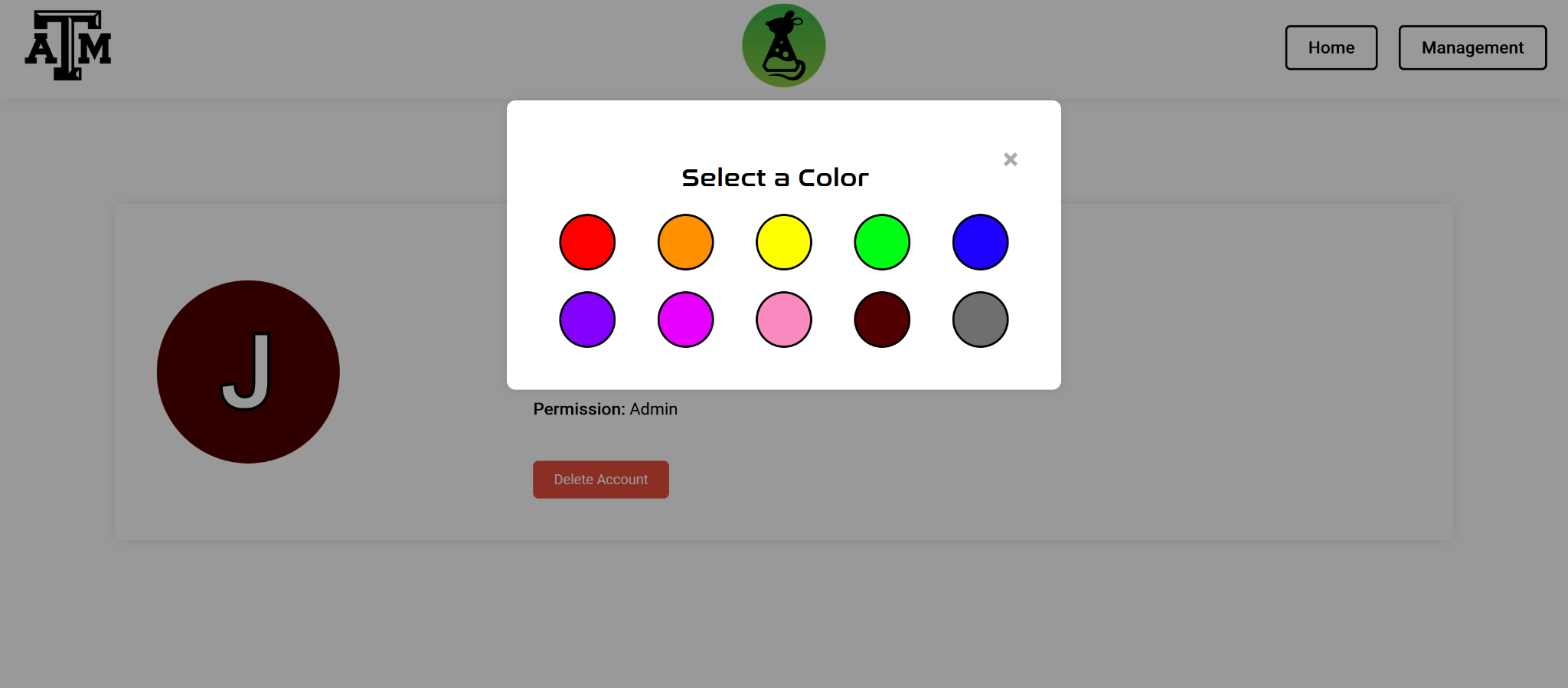
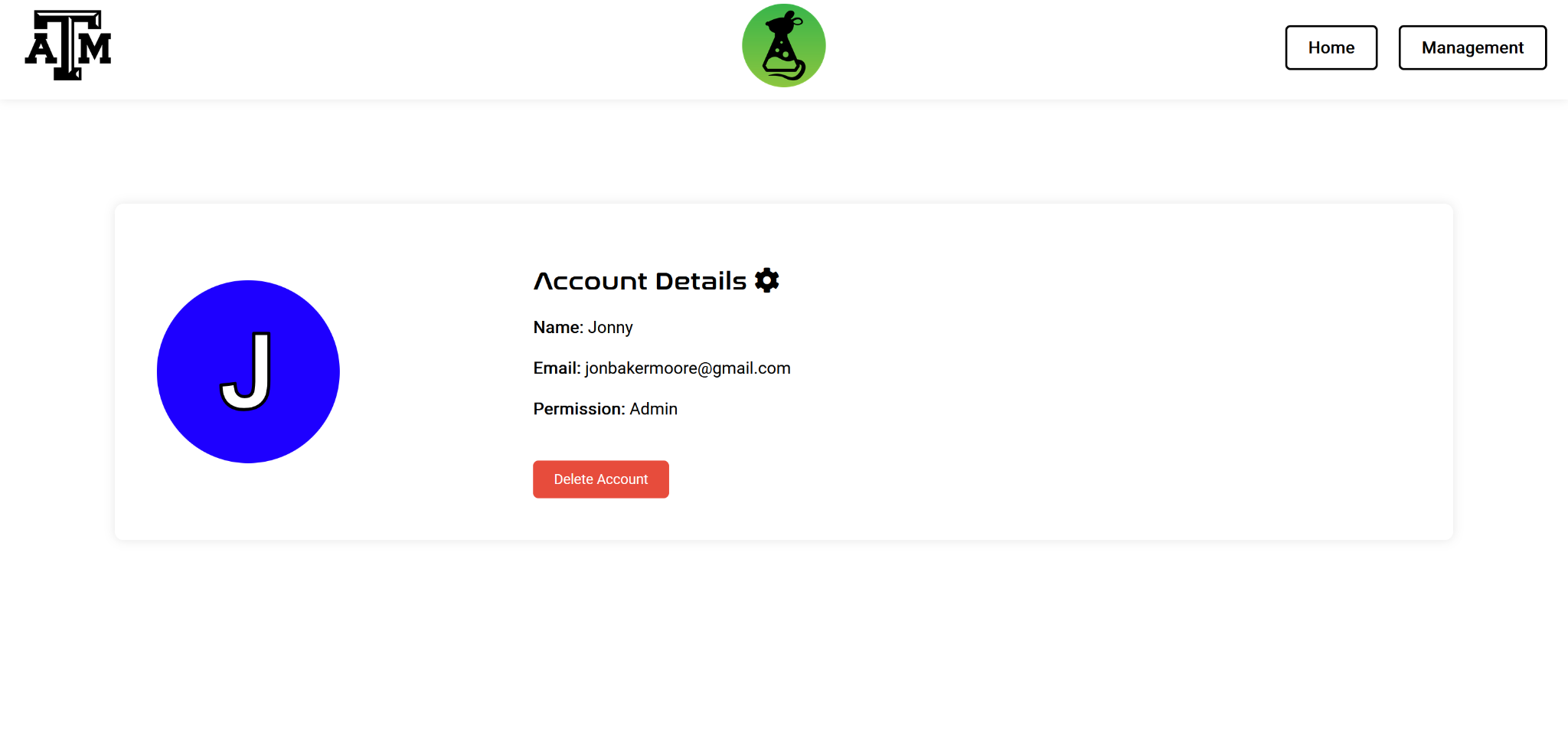


**Figure 11: Account Page**

****

**Figure 12: Edit Account Modal**

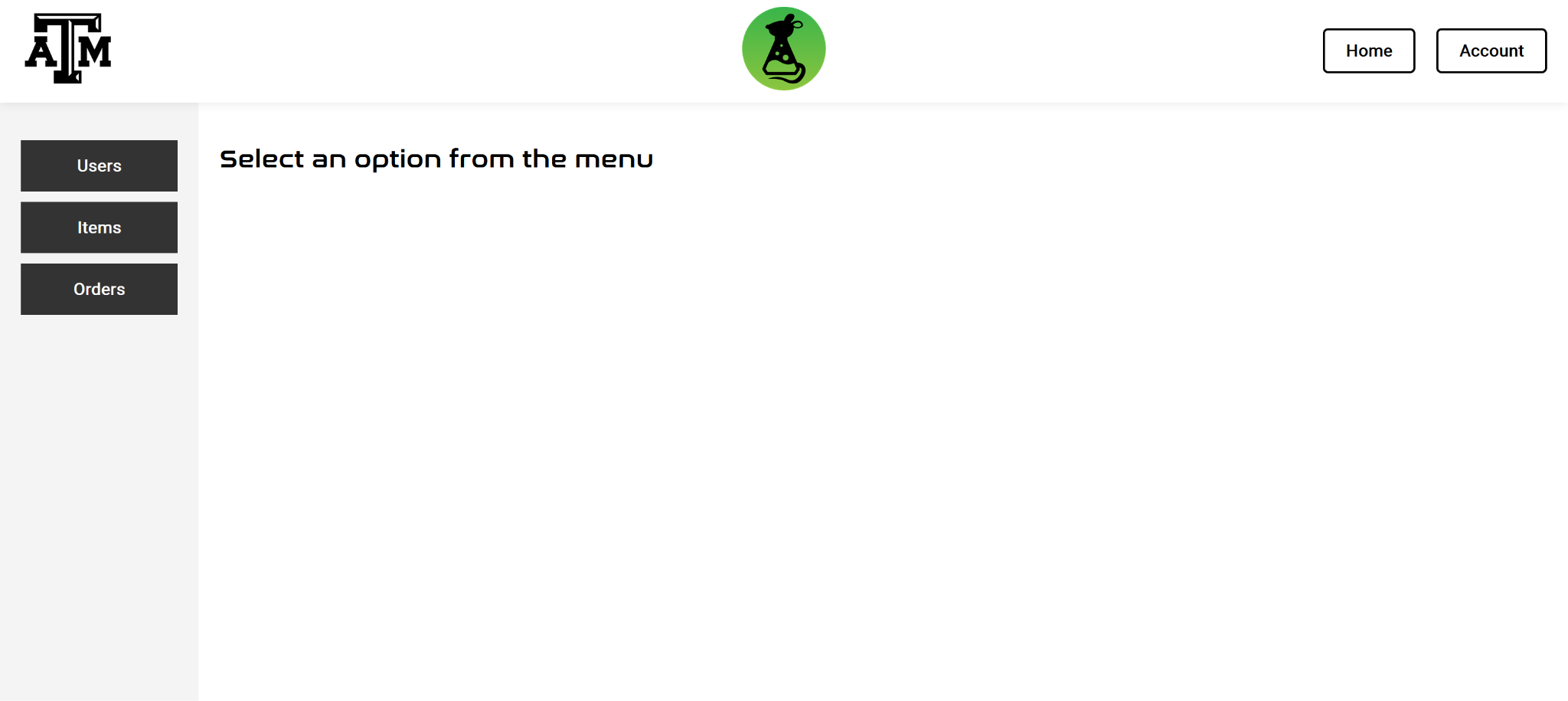
There is also another unnecessary but fun addition of profile color customization. As you can see in Figure 13, a color modal appears after clicking on the user icon. Clicking a color automatically changes this for the user. The only way to know this is a feature is to hover over the user’s initial, so it is more of a fun Easter egg for people to discover.

**Figure 13: Color Modal Figure 14: Color After Selection**

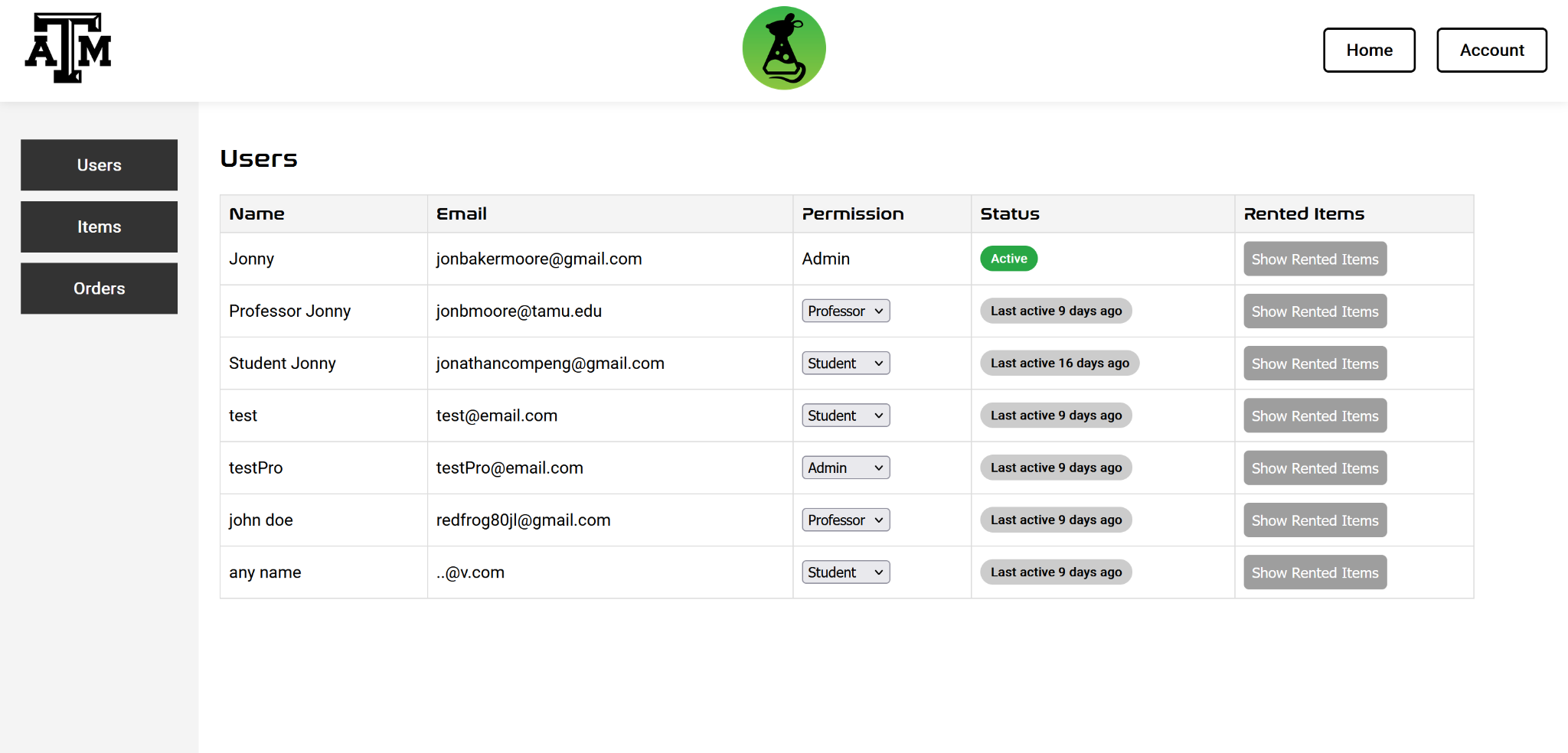
**5.5 Management**

There is also the matter of managing users and data behind the scenes. For accounts that have “Professor” or “Admin” status, the management page is available. It is accessed through the account page or through the user icon dropdown menu on the home page. From here, there are three options: Users, Items, and Orders.



**Figure 15: Management**

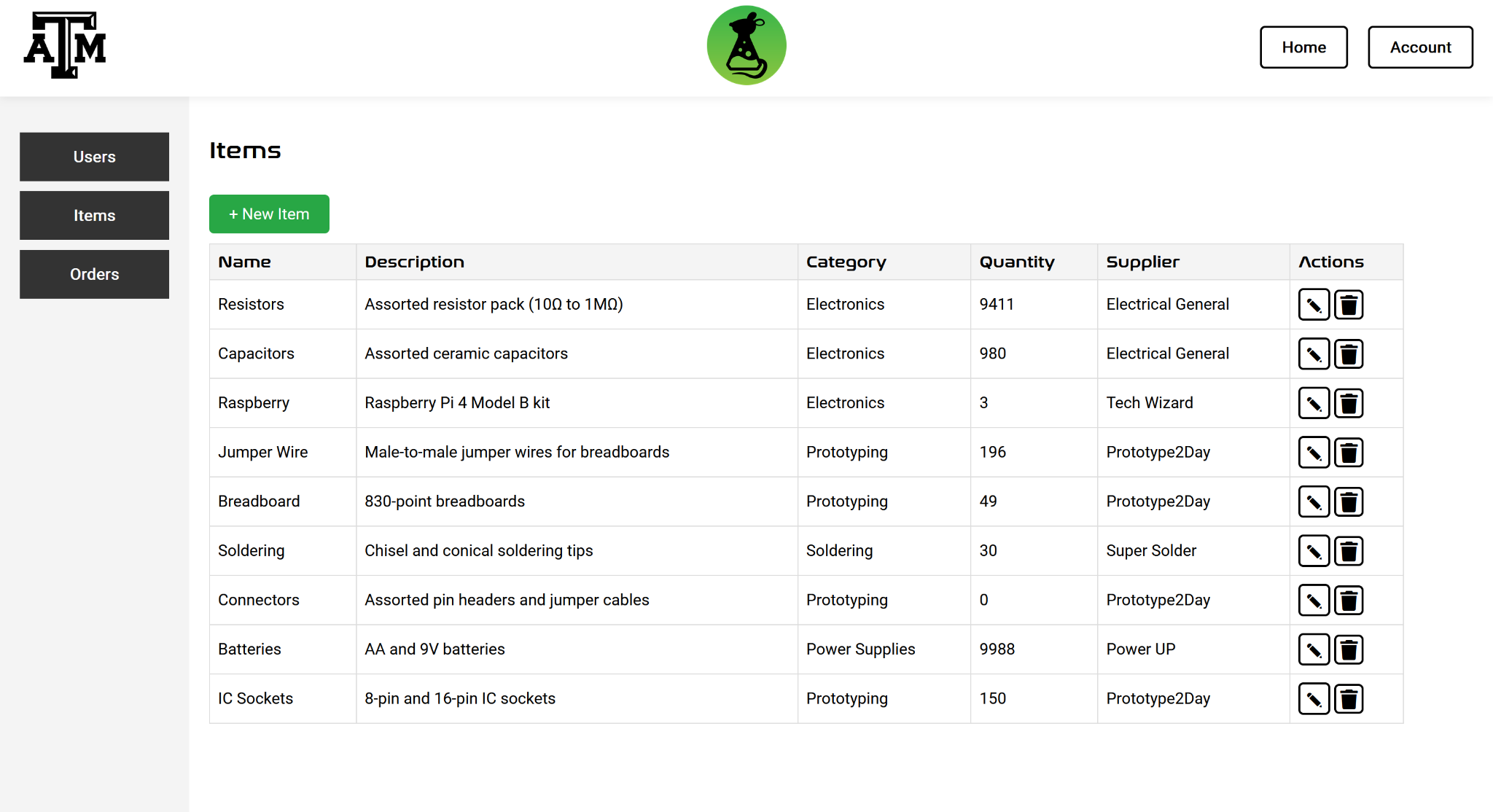
Clicking on these options on the left displays a different table of information. From the Users tab, professors and administrators can see all the registered users, their names, emails, and the last time they were active. In Figure 16, there are a lot of fake users mostly for testing, so please ignore it.



**Figure 16: Users Table**

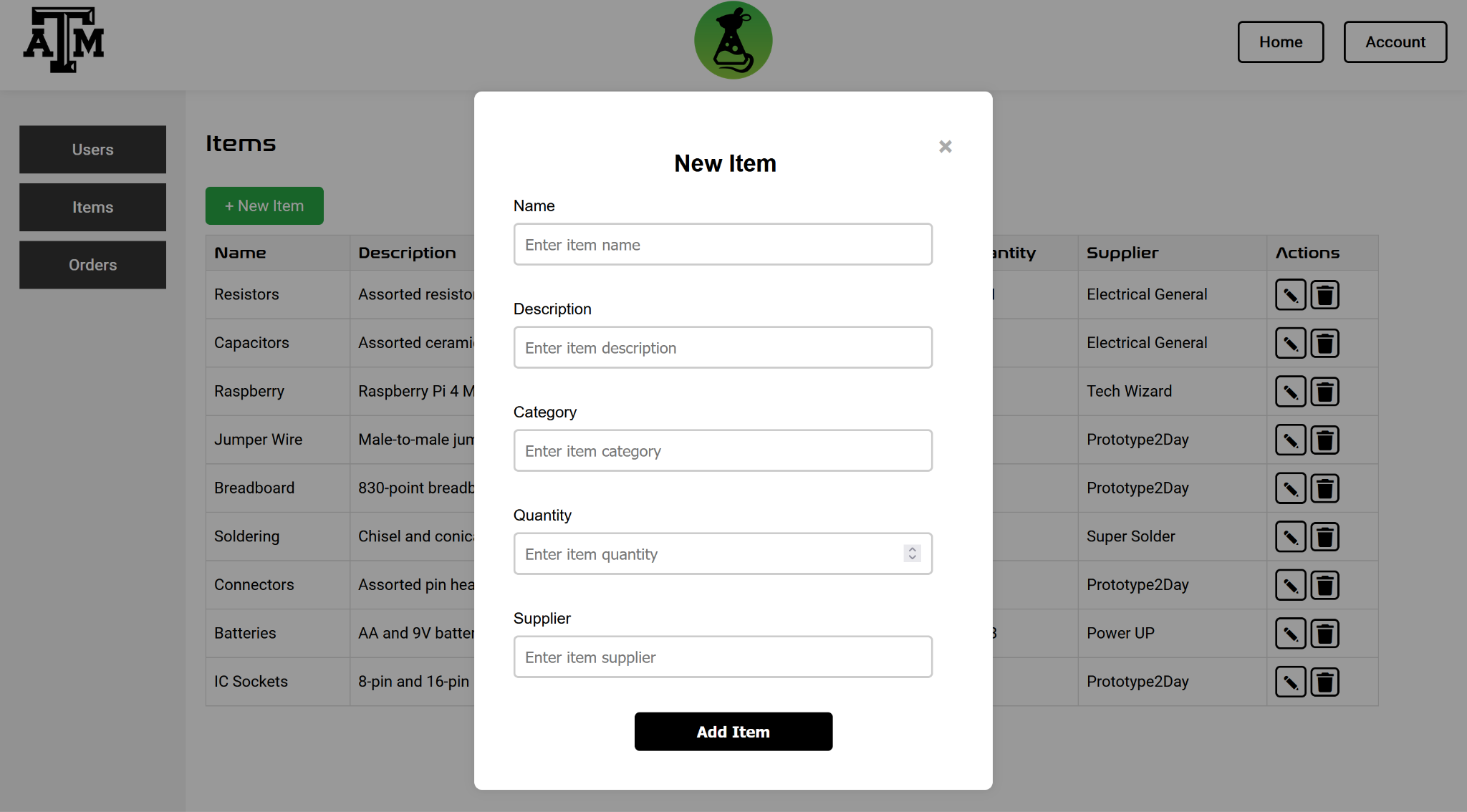
From this page, only admins can change the permission level of users, so if someone does not meet the requirements, the column does not appear. Also in the table, the rightmost column has a button that lists all items that user is currently borrowing from the lab.

Below is the Items table. From this table, all items are listed, and on the rightmost column there are icons for editing and deleting items. Above the table there is also a button for adding items.



**Figure 17: Items Table**

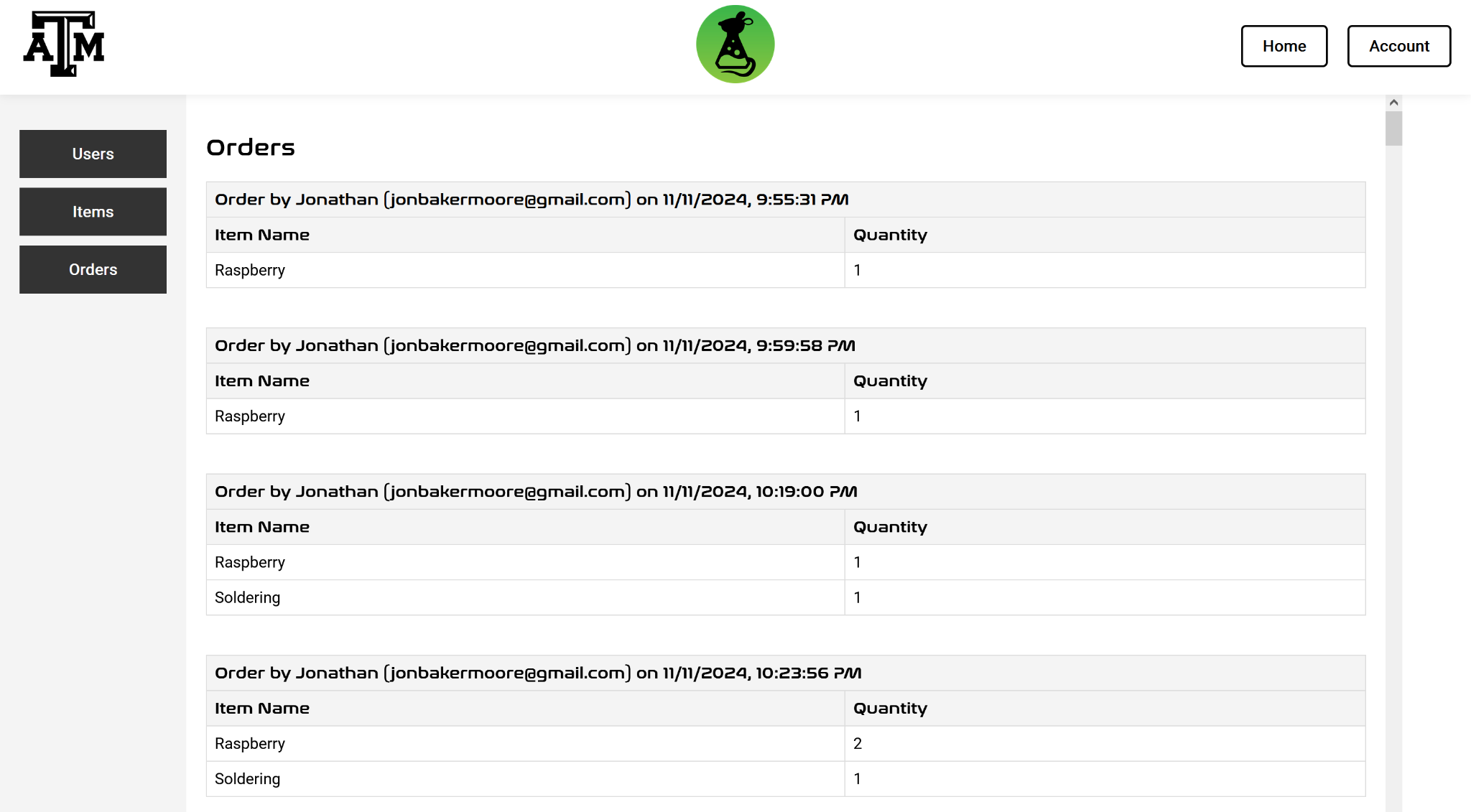
The button for new items and the edit (pen) icon both create a modal with a form for the user to fill out. All fields are required for new items, but editing items can be done by only changing a single attribute and submitting. For simplicity, Figure 19 is for new items, but the format is extremely similar for editing items.

****

**Figure 18: New Item Modal**

When the lab receives more of an item, the idea is that professors or administrators go to this page to change the stock of that item. When new items are ordered, Figure 19 is what admins need to fill out to add them.

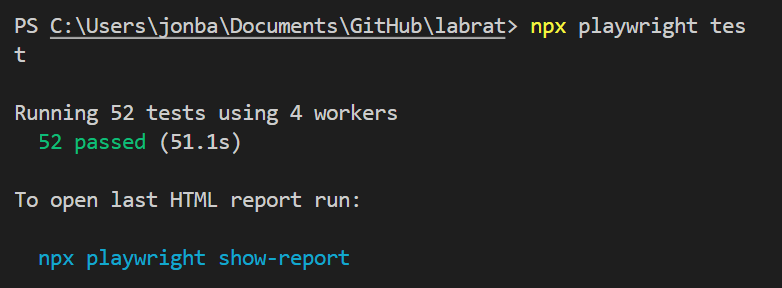
Finally, there is the orders page. Orders are listed in order of time and have headers listing who ordered the items, what their email address is, and when they ordered. Orders with multiple items have all of the items listed together and have each quantity listed.

****

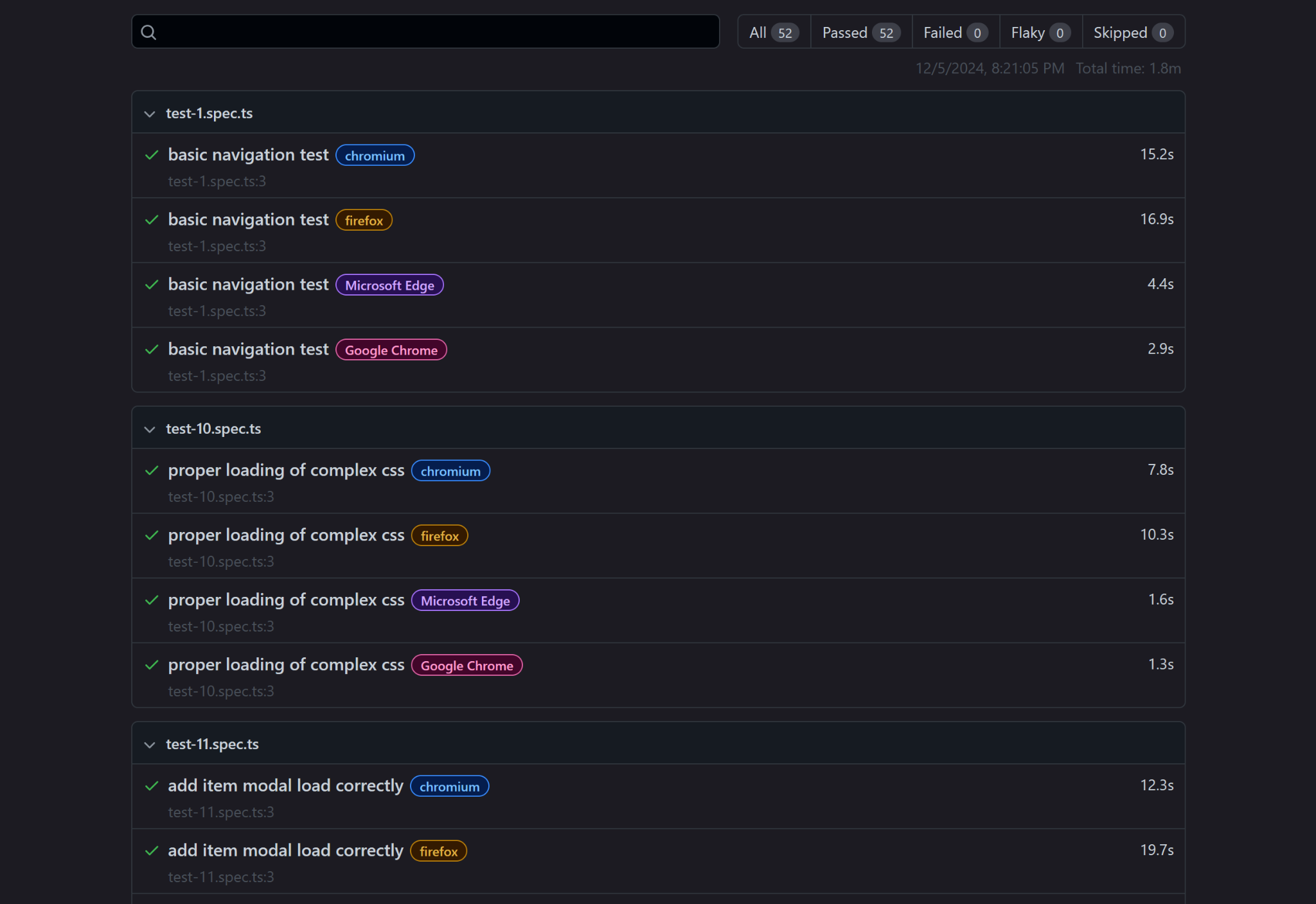
**Figure 19: Orders Table**

**6. Validation and Testing**

Validation of this website is fairly straightforward. Using a testing library called Playwright, I am able to record interactions with the website and have those inputs tested across many different browser types all at the same time. I have many tests written, all to help with regression testing for changes between browsers and all fit the validation plan steps that I wrote at the beginning of the project. Below is the terminal output after running the tests followed by a snippet of the HTML file that reads what tests were performed and how long each one took.



**Figure 20: Playwright Terminal Output**

****

**Figure 21: HTML Output**

**7. Challenges**

Project went very smoothly. Hardly ran into issues, as my previous semester had a class over web development. Lucky to say that I did not experience many issues during development. Any issues were resolved very quickly once I consulted Mozilla’s Javascript documentation.

**8. Integration Plans**

The only left to do when integration begins is get the database hooked up properly and troubleshoot any bugs that may appear. It should be a very short process when the time comes in ECEN 404. I also need to host the website, and I have chosen a service called Render; I chose Render for its Node.js server support.

**9. Future Improvements**

Aesthetically, there are a few things I would like to change about the search bar, filters, and items table on the home page after login. I don’t like the bubbly style I went with so I will likely be changing that soon.

**10. Conclusion**

The web application is ready for launch. Once I get it hosted online next semester and integrated with our database, most - if not all - requirements will be complete. It serves its purpose of being a tool for students and faculty alike. Hopefully, this can help solve Texas A&M’s laboratory organization issues.

**11. References**

Mozilla: mdn web docs | Javascript; https://developer.mozilla.org/en-US/docs/Web/JavaScript