

The Stock'Em Team  
CSCE 606 Software Engineering  
Sprint Plan #4

**The Team –**

*Mark Sturtevant - PRODUCT OWNER*

*Alisha Raj*

*Vinay Singh*

*Simran Kaur - SCRUM MASTER*

*Simran Jot Kaur*

*Tanishq Chopra*

*Ishaan Rawal*

**Customer Meeting Date/Time –**

Contact: Paul Taele

Current Meeting Times: Every Wednesday 1:30 pm - 2:00 pm, Zoom

**Summary –**

Our target audience/stakeholders are CSCE capstone professors who desire a new web service to simplify the process of managing the inventory of expensive items that will be checked out by students as they do their projects. Such a solution to this involves managing a database of items to (a) give all relevant information and history about an item, (b) tracking which items are checked out by which students, and (c) having an easy-to-understand AI to allow for this information to be quickly updated. In doing so, security is a top priority as only professors or verified TAs should be able to authenticate these assignments of items.

Our application will meet these needs by utilizing Ruby on Rails to achieve the three points listed above by incorporating a database model with a frontend UI, all while being authenticated by an SSO to achieve the three points above whilst maintaining security. As time permits, we can further

embellish this project by giving support for student inventory requests or notifying professors when relevant inventory items have not been returned on time.

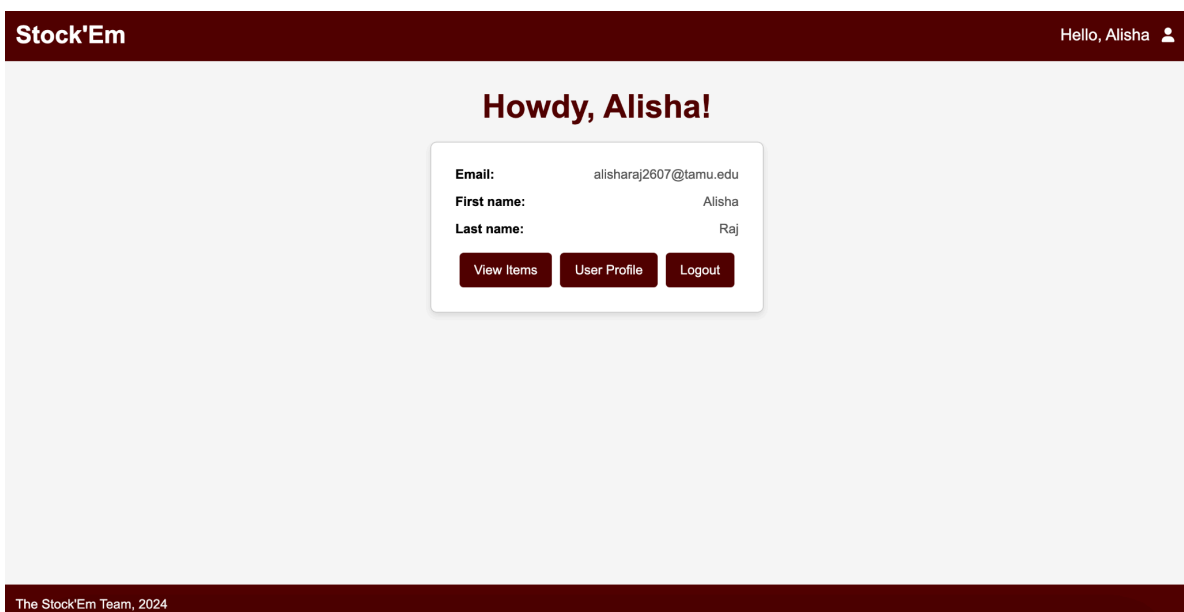
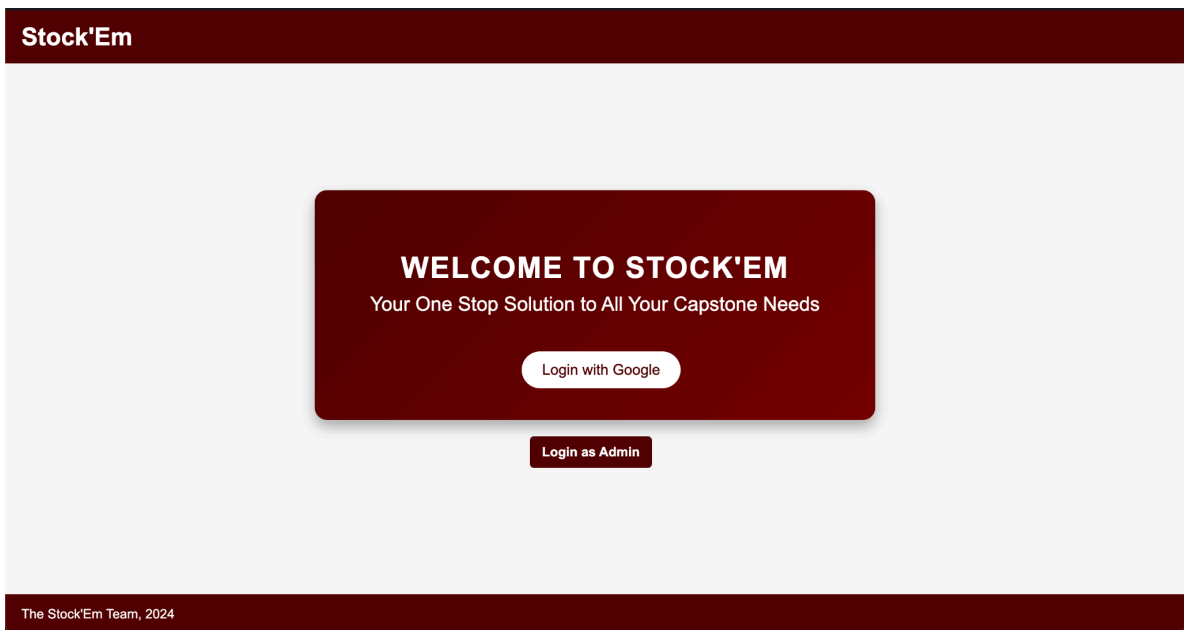
## **User Stories –**

- {Bug Fix} Redundant Column in the users table should be removed
- {Bug Fix} Merge Status and Availability Fields into one
- {Bug Fix} Fix the search filter bug
- [UI] Make pages responsive and fix all the navigation issues
- [API/UI] Users should be able to edit and delete an individual item.
- [API/UI] Any updates, edits to an item should be tagged as a item history in the events table
- [UI] Fix all the CSS issues, optimize the styling (create central application css for the whole system)
- [API/UI] Implement more advanced search functionality.
- [API/UI] User should be able to check in or checkout items
- [API/UI] Set up rules and procedures for authorization in the system
- [API/UI] Users should be able to create a new item from the home page.
- [API/UI] Users should be able to add notes/comments to an existing/new item in the item details page.
- [API/UI] Users should be able to view added notes on the item details page.
- [API/UI] Users should be able to add/edit details to his profile.
- [API/UI] Users should be able to view and use the search filter in the items list page.
- {Bug Fix} User details should be populated correctly in the user profile page.
- [STRETCH] I am a professor who wishes to obtain an item from inventory for one of my capstone groups.
- [STRETCH] I am a professor who wishes to obtain an item from inventory for one of my capstone groups.
- [STRETCH] Users should be able to change attributes(damaged/lost) of a particular item.

- I'm a new user, I need to create an account (sign-up) using google account(mp) and then I should be able to login.
- I'm a global admin, I should be able to login to the app (with some predefined credentials or google account and then get authorized.)
- [STRETCH] I'm a global admin, I should be able to authorize new accounts and assign roles.
- I am a facilities manager who wishes to know where every item in inventory can be found.
- I am a facilities manager who is relocating many inventory items from one building to another and want some means to record this.
- I am a professor who wishes to know the wear-and-tear on an item before I check it out for my students.
- I am a professor who needs to know where to find an item in inventory when I check it out.
- I am a facilities manager who needs to know who to contact when a capstone team has not returned their item in time.
- I am a student who has lost an item and need some way to report this.
- I am a professor who wishes to add a special note about an item such as how it should be used, important details, or more.
- I am a facilities manager who wants some method to determine if an item in inventory needs to be replaced.
- [STRETCH] I am a student who has just damaged an item and need some way to report this.
- I am a professor who wishes to know which items of a certain category (i.e. smartphones) are available for checkout.
- I am a senior Capstone professor who wants to ensure that all inventory updates must go through verified professors.
- I am a facilities manager who wants the ability to quickly add and/or export a large number of items onto/from the website.

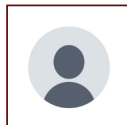
## **Current Progress:**

*Home Page*



*User Profile Page*

## USER PROFILE



**Name:** Alisha Raj  
**UIN:** 1234  
**Email:** alisharaj2607@tamu.edu  
**Contact No:**  
**Role:**  
**Status:**



## View Items

All Categories 

All Statuses 

**Filter**

☐ Only show available items



**multimeter**  
Electronics  
**Not Available**  
Status: No Status



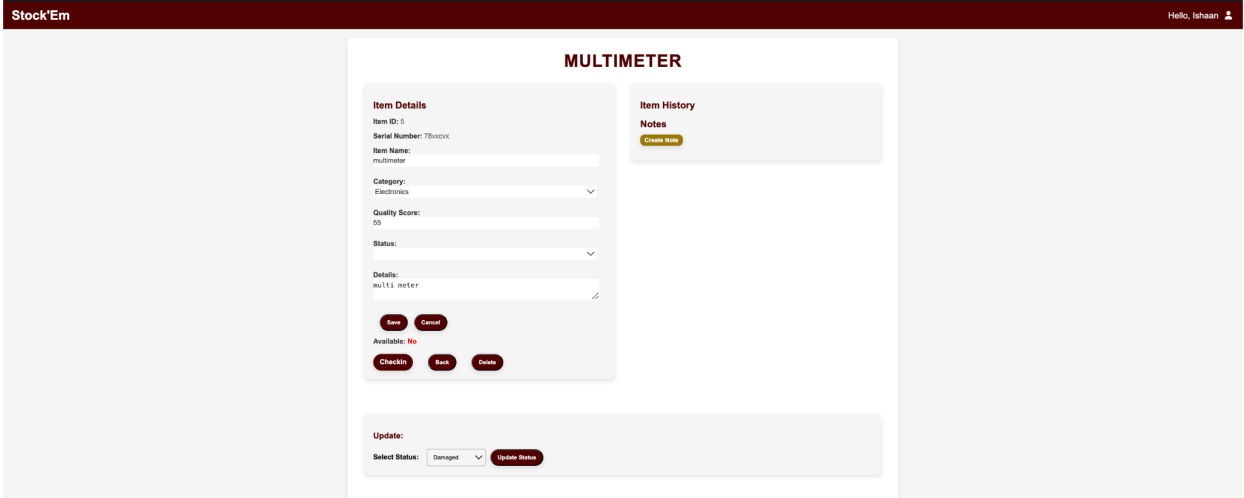
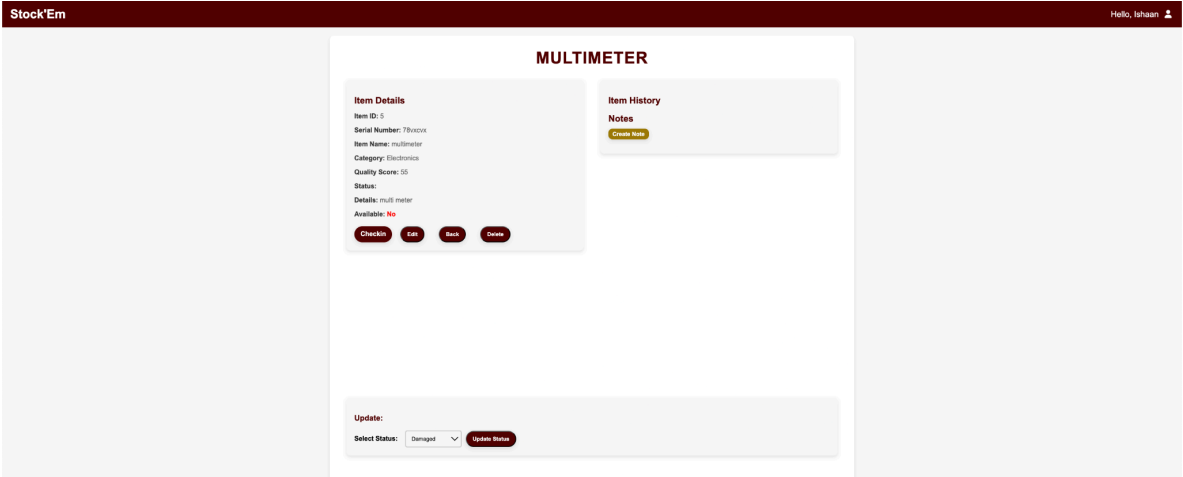
**PCB Kit 289**  
Development Boards and Kits  
**Available**  
Status: Damaged



**iphone8**  
Mobile Devices  
**Available**

Create New Item

## Item



Create New Item

The screenshot shows a web application interface for 'Stock'Em'. At the top, there's a dark red header with the text 'Stock'Em' on the left and 'Hello, Alisha' with a user icon on the right. The main content area is light gray and contains a white card titled 'Create New Item'. Inside the card, there are several input fields: 'Item Name' (text input), 'Category' (dropdown menu), 'Quality Score' (text input), 'Currently Available' (dropdown menu), 'Serial Number' (text input), and 'Details' (text input). At the bottom of the card, there is a red button labeled 'Create Item'.

## Sprint 4 Plan

*Sprint Goal: The goal of this sprint is to (1) privilege support (admin/student etc) (2) Improve the UI for better user experience. The system should support users to create, modify, check-in and check-out the items based on their assigned privileges. Fix all the bugs found till date and add any remaining QoL features.*

### Tasks:

- [UI] Hide Buttons that user is not privileged to use (and/or give a flash that said operation is not supported) [5] **SIMRAN J**
- [API] Controller Security: Block operations from unprivileged users [5] **ALISHA**
- {Bug Fix} Fix unknown status dropdown search filter [1] ALISHA
- [UI] Events & Notes: Improve formatting to look nicer and clearer [3] **TANISHQ**
- [UI] Make History and Notes Section Separate [2] **TANISHQ**
- {Bug Fix} Validation for Update Item Status: if item is already that status, don't create an event [1] **ISHAAN**
- [API/UI] Dropdown for locations instead of manually typing locations in, perhaps with an "other" option as well [2] **ISHAAN**

- [UI] Illegal Field Validations for all the fields in the app [3] **ISHAAN**
- {Bug Fix} Fix Checkout/Checkin Button Formatting on Item View Page [1] **VINAY**
- [UI] Change formatting when editing an item (instead of inline, do popup, new page [4] **VINAY**
- {Bug Fix} Fix Create Item Button disappearing on some screens [1] **VINAY**
- {Bug Fix} When you filter out all items, display a message “No Items” [1] **SIMRAN**
- {Bug Fix} Make flash messages more consistent (in terms of fading out vs not fading out) [2] **SIMRAN**
- [UI] Warning Popup for Delete Item [1] **SIMRAN**
- [UI] Auto-Filter when updating a dropdown on View Items Page [1] **SIMRAN**
- [API] Create a script and UI for adding and exporting items in bulk [5] **MARK**

Total: 15 Tasks with 37 Points

### **Sprint 3 Progress: [COMPLETED]**

*Sprint Goal : Users should be able to check in, check out items and view related history. System should support editing and deletion of the item and provide more advanced search features. User experience should be enhanced to support better navigation. Fix all the bugs found till date.*

Tasks :

- {Bug Fix} Redundant Column in the users table should be removed [4] **SIMRAN**
- {Bug Fix} Merge Status and Availability Fields into one **TANISHQ**
- {Bug Fix} Fix the search filter bug [3] **SIMRAN**
- [UI] Make pages responsive and fix all the navigation issues [5] **TANISHQ**
- [API/UI] Users should be able to edit, delete or relocate an individual item [5] **ISHAAN**



- [API/UI] Any updates, edits to an item should be tagged as a item history in the events table [3] **VINAY**
- [UI] Fix all the CSS issues, optimize the styling (create central application css for the whole system) [8] **SIMRAN J**
- [API/UI] Implement more advanced search functionality. [6] **ALISHA**
- [API/UI] Users should be able to check in or checkout items along with viewing the history of an item [5] **MARK**
- [API/UI] Set up rules and procedures for authorization in the system [5] **VINAY**

## **Sprint 2 Progress: [COMPLETED]**

*Sprint Goal: The goal of this sprint is to build on the existing functionalities and enable users to add, update and view the item data on Stock'Em. Some stretch goals include authorization after logging in to ensure only relevant information is displayed and profile is updated before performing any other operations.*

### Tasks:

- User Story: Users should be able to create new items from the home page. [5] **SIMRAN J**
- User Story: Users should be able to add notes to an existing/new item in the item details page.[5] -done **MARK**
- User Story: Users should be able to view added notes on the item details page[3]-done **VINAY**
- User Story: Users should be able to add/edit details to his profile[3]-done **SIMRAN**
- User Story: Users should be able to use a search filter in the items list page.[5] **ISHAAN**
- User Story: Users should be able to change attributes(damaged/lost) of a particular item[3] **VINAY**
- User Story: Users want a clear, formatted information page describing an item.-done [5] **ALISHA** (+notes formatting)

## **Sprint 1 Progress: [COMPLETED]**

- *User Story: I am a professor who wants to be able to store a large list of fields for each item, including item details, history, responsible persons, etc. [Infra]*
- Database UML Diagram **ISHAAN**
  - Turn Database Design into Official UML diagram
  - (maybe) some other UML diagrams too
- *User Story: I am a user of the Capstone Inventory Management service who wants to ensure no bugs will occur while I'm using the system.*
- Database & Authentication Tests Cucumber Tests **ALISHA**
  - No view / app interaction yet; just test that certain fields exist in the database
- *User Story: I am a professor who wishes for a hosted application that I can use to manage Capstone inventory.*
- App Creation & Database Migrations **VINAY**
  - Will be the first push to the github to create the rails app
  - Add the database migrations
  - Heroku Creation and Migration
- *User Story: I'm a new user, I need to create an account (sign-up) using google account(mp) and then I should be able to login. [Backend]*
- Authentication **SIMRAN J**
  - Google SSO Authentication Support
- *User Story: I am a facilities manager who wishes to know all of the information about an item, including location data, who to contact, and any item history.*
- Frontend: Item Information Page **MARK**
- *User Story: I am a professor who wants to be able to see all available items for checkout at once.*
- Frontend: Home / All Items Page **SIMRAN**

- *User Story: I'm a new user, I need to create an account (sign-up) using google account(mp) and then I should be able to login.*  
*[Frontend]*
- Frontend: Login Page **TANISHQ**

## **Links –**

GitHub Repo: <https://github.com/LOGiC31/Stock-Em>

Ticket Management: <https://github.com/users/LOGiC31/projects/1>

Slack: <https://tamu.slack.com/archives/C07PGFSLJV8>

Heroku App Link: <https://stockem-0caaa6588e50.herokuapp.com/>