

Criterion E: Evaluation

Evaluation of the product

SUCCESS CRITERION	MET?	COMMENT FROM {CLIENT} ¹
The program should have clear and viewable three functions 1st - for {Client} to create her inventory of supplies; 2nd - for {Client} to create a lab and its requirements; 3rd - for {Client} to determine if she can perform the lab(s) based on inventory supplies	✓	“The application allows me to create a table for my inventory supplies, create labs for these supplies, and predict their possibility”.
The program allows {Client} to view her inventory of supplies based on the name of the supply and type of supply	✓	“From testing, I can view the inventory whenever necessary and can see each item’s stats”
The program should allow for inserting items into the inventory of supplies	✓	“Adding items works perfectly”
The program should allow for deleting items from her inventory that is already pre-existing	✓	“Removing items works”
The program allows for {Client} to view a number of already created common labs of her subject type	✓	“I already see that you’ve created some labs for my classes”
The program allows for {Client} to create her own personalized lab activities	✓	“When I click on the Labs section I can create new labs with my own requirements”

¹ Refer to Appendix E.1 for the full interaction

with her own supply count and types of item		
The program allows for all these item counts for Inventory to be displayed on a database table for viewing pleasure	✓	“The Inventory function allows me to view the type and number of items from clicking ‘display’ button”
The program allows for {Client} to update any count values of the items in her inventory based on the name	✓	“I don’t have to always remove an item to change its value, I can click the ‘Update’ button, letting me edit its frequency”.
The program allows {Client} to calculate if the lab or labs she chooses are possible to perform based on her current amounts of supplies in her inventory	✓	“If I go over to the Calculate section of the application, I can choose any of my Labs and press the ‘Calculate’ button and see whether I can do this lab, or I need to restock inventory”
The program allows for all of these functions and results to be displayed using Swing (graphical user interface)	✓	“I can find everything easily because the visual elements of the application are laid out well”.
The program information for helping and guiding {Client} in case she gets stuck	✓	“Your application is relatively easy to follow and the descriptions/help blurbs are useful”
In the case of database errors, informational messages in GUI will appear on the screen	✓	“If something in the database messes up, like installing, it lets me know”.

The program works fully and yields accurate results when used correctly. All aspects have been tested extensively by myself and {Client} and {Client} believes that the program “meets and exceeds her expectations” and “solves the problem well”.

Extending the Application

1. Listing restocked

{Client} and I felt that the feature of listing exactly **what** items need to be restocked² if a lab **cannot** be performed, was needed. The feature would be coupled with the Output in the Calculate window:

Create a New Lab				Choose from already created Labs			
<input type="button" value="Back"/>	Item for New Lab: <input type="text"/>	Item Count for New Lab: <input type="text"/>	<input type="button" value="Submit Items"/>	Output:	<input type="radio"/> CELLLAB	<input type="radio"/> ONIONLAB	<input type="button" value="Calculate"/>

Adding this feature would increase work efficiency³ for {Client} – they wouldn't need to switch back to Inventory table to check what they're missing – instead just rely on the output and restock accordingly.

Create a New Lab				Choose from already created Labs			
<input type="button" value="Back"/>	Item for New Lab: <input type="text"/>	Item Count for New Lab: <input type="text"/>	<input type="button" value="Submit Items"/>	Output:	<input type="radio"/> CELLLAB	<input type="radio"/> ONIONLAB	<input type="button" value="Calculate"/>
Sorry, this lab cannot be performed – please restock inventory with Item: _____ with count of _____							

² Refer to Appendix E.2 for verbatim

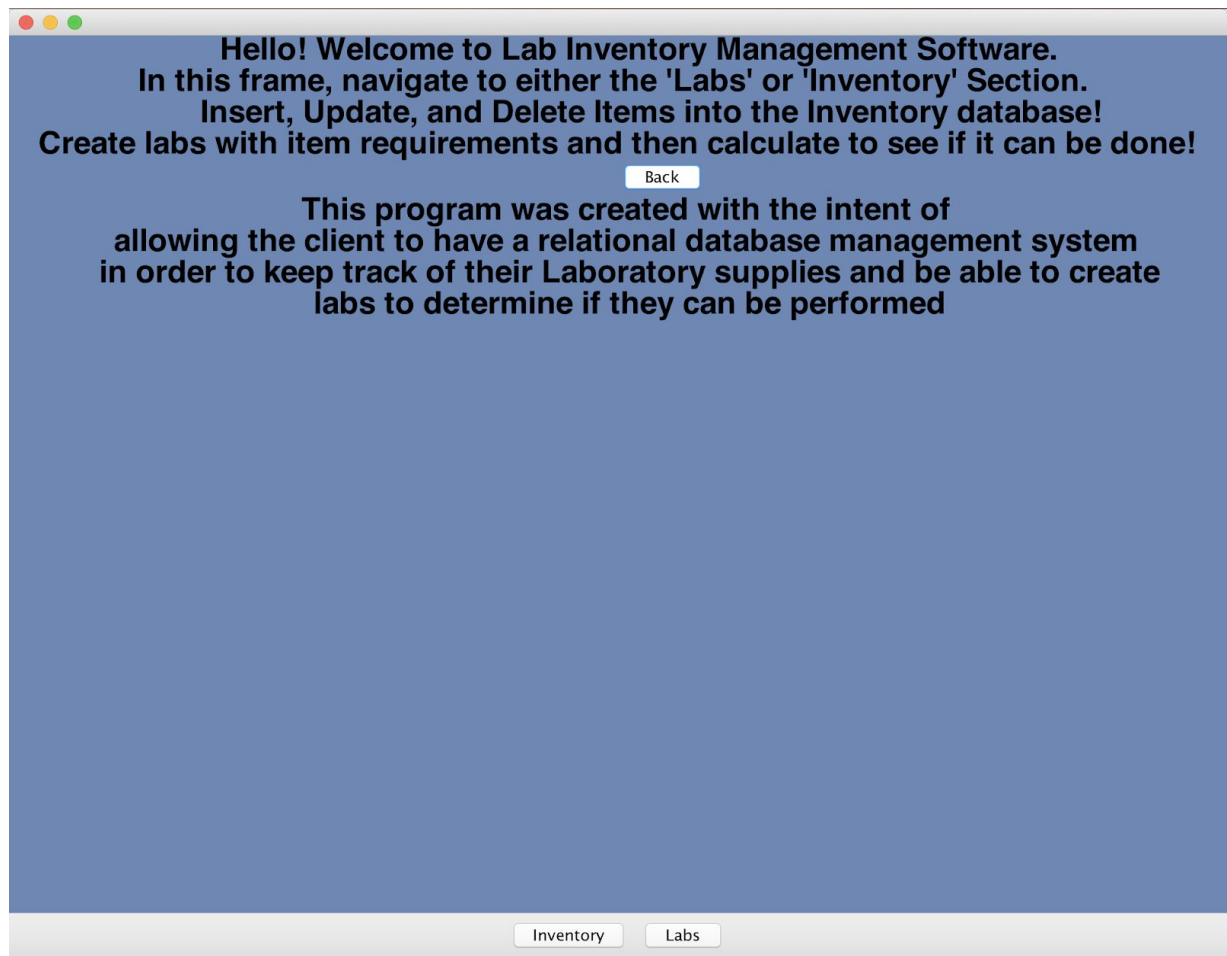
³IBID

2. Subjects Split

{Client} and I agreed to have the option to “**create a new database for inventory based on subject**”⁴ – “**one Biology one for Anatomy**”.

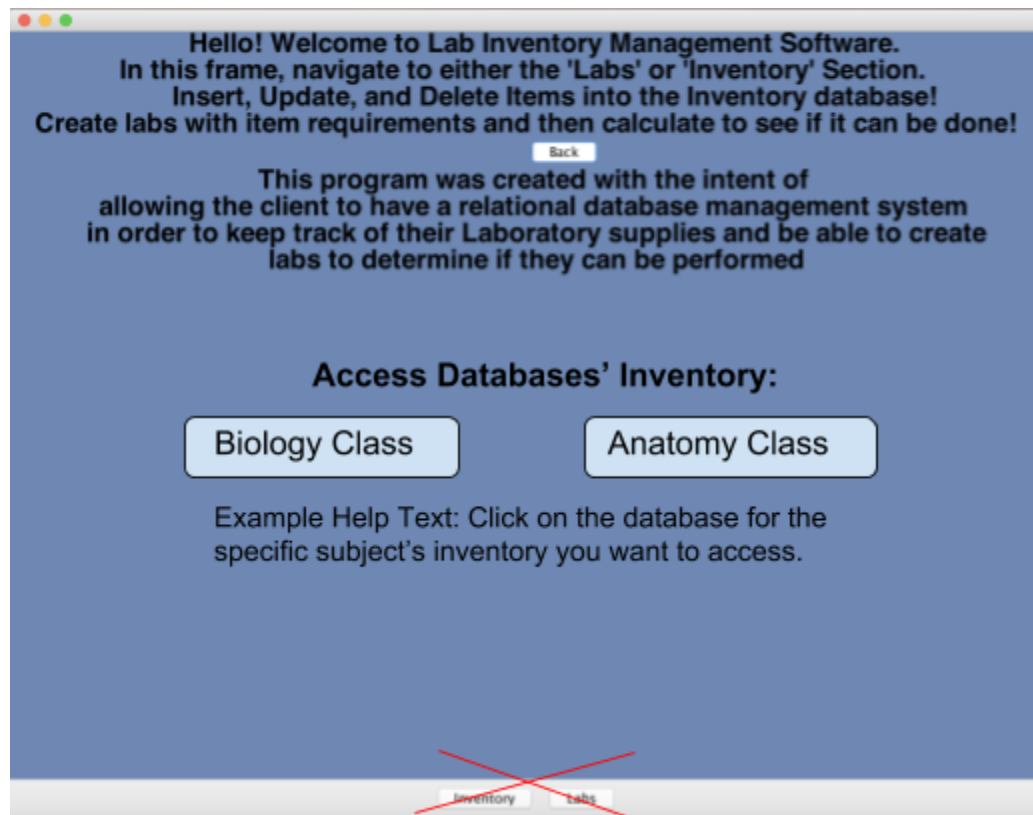
These would be **presented in class HomeFrame.java**⁵.

Additionally, {Client} could “**choose**” which database to **access**, which would lead her to a different Inventory of supplies and different Labs here:



⁴IBID

⁵ IBID



Adding this would **“increase organization”** for {Client} as the splitting into subjects means that she can **“better enter supplies and labs”** and need not **“sift through irrelevant data values”**⁶ for supplies and labs.

⁶ IBID