1/19/11
Researched shopping carts and how they work:
- To access database, go to localhost/phpmyadmin/
- Product listings are stored in a table ("items", under "cart" database)
- "Cart" table stores one entry for each item in a particular user’s shopping cart
- Need to write php files for pages that connect to the database. For example, product listings page must be a php file because it must display all the products in the "items" table
- Since many pages will need to interface with the database, I should create a php file called db.php with the following:
  - Variables for server, user, password, db name
  - Functions: connectToDB(...), getCartID(...)

1/22/2011
Researched Checkout options:
- Could use a certified PCI-compliant checkout system; is this a payment gateway?
- Google Checkout has code to tell me about purchases (bookmarked)
- Will worry about this later

1/29
Created Home, Products, and FAQ page on Joomla
- Modified top menu to include these 3 pages
- Made home page the default home page
- Removed unnecessary modules from home page

Researched how to remove/move main menu (since I’m using the top menu)
- Did this by copying the top menu links into the main menu, removing the top menu (using the module manager), and setting the main menu position to “user3”

Realized I need to have a blog, so I started researching how to implement blog functionality
Can’t get Brotankery logo to appear on top-left of page… got it!
- Image needs to be in /templates/rhuk_milkyway/images (not /images)
- Still need to mess with CSS to get header looking correct, but will do this later

Created a blog
- Created a Section and Category, both called Blog
- Whenever I want to add a post, create an article and list it under the Blog section/category.
- The settings for the Blog category are such that newest articles appear first

Issues to resolve
Header: need to get rid of superfluous horizontal rule, need to get main menu to line up with divider
General: need to get simpler permalinks

2/11
Downloaded Dreamweaver, synced it with Brotankery site. Now I can edit the pages from Joomla and from Dreamweaver.
Things I did today:

- Created home page
- Created header and footer php files, to dynamically load header/footer
- Created products database, which holds product ID, name, description, price, and category
- Created admin section, which allows admins to log on and add items
  - Admin section uses cookies to store sessions
- Created item edit section, which allows admins to edit preexisting items

2/20
Things I did today:

- Updated admin session cookies to store MD5 hash of password, not password itself
- Created product_page.php, which displays all the products using a php function displayProduct()
- Created an inventory database to hold inventory of XS, S, M, L, XL of each product
- Created a drop-down menu in displayProduct() that displays on the sizes in stock
- Created inventory_edit.php in the store-admin section, which allows admins to edit inventory easily
- Spent 1 hour debugging a stupid SQL error: “UPDATE <db> SET <field> WHERE <condition>” didn’t work because WHERE needs to go at the end, with NO commas before it

To do next week:

- Add “Add to cart” button in displayProduct()
- Create cart using php
- Display cart

3/6
Things I did today:

- Created an “Add to cart” button for each product
- Created cart.php to display (primitively) items added to cart

Need to fix:

- Product ID and Size variables not sent properly to cart.php. I suspect this is because each cart button is rendered by the same php function...

3/8
Things I did today:

- Fixed the “add to cart” button (product id was already being sent correctly; fixed size variable by moving the select options dropdown menu for size variations INSIDE the form for “add to cart”
- Studied websites with cool designs
- Drafted design for site layout (background, content interaction, accordion-menus using AJAX)

3/9
Things I did today:

- Formatted shopping cart to display item name, quantity, price, and total price
- Created delete button to remove items from cart
- Created quantity input field for user to modify quantity
- Made quantity input field safe against invalid/dangerous inputs
- Ensured quantity field cannot exceed current inventory
• Added Google Checkout button: takes user to Google Checkout to pay for items in cart
• Tested Google Checkout button – works!
• Changed Google Checkout button to Sandbox mode
• Researched Google’s Notification and Order Processing API
• Studied well-designed websites for ideas

To do later:
• Implement Google Notification and Order Processing API integration to record transactions and update inventory
• Add blog and faq page
• Add contact page (for special requests)
• Design:
  o Header (no reload)
  o Nav bar (no reload)
  o Background image (no reload)
  o Footer (no reload)
  o Individual pages: shop, product pages, blog, faq, contact, cart (if not integrated into shop)

3/10
Things I did today
• Added FAQ page
• Decided that blog can wait until later (not essential)
• Studied well-designed website for ideas
• Now: design

3/11
Things I did today
• Designed 2 drafts of homepage
• Learned photoshop essentials

Things I did later today
• Got Google Checkout sample code from Google
• Created 2 new tables: transactions and transaction_items (to record items bought)
• Modified responsehandler.php to update transactions table when receiving a new-order-notification from Google

To do tomorrow:
• Modify responsehandler.php to update transaction_items table and to update inventory table
• Enhance transactions table to store more fields
• Create transactions dashboard that displays all fields in transactions and transaction_items tables, and allows admin to edit, as well as charge/ship/archive

3/12
Things I did today:
• Created function to update transaction_items table
• Created function to update inventory table
- Moved code around for better modularity/organization (created inventory_functions.php, size_functions.php, and recordtransaction.php)
- Decided to keep admin dashboard simple (not to add features listed yesterday)

To do later:
- More features (including special order request form)
- Design
- Testing
- Host website on production server

3/15
Things I did today:
- Added new field to transactions table called financial_order_state (REVIEWING, CHARGING, CHARGED, CANCELLED, CANCELLED-BY-GOOGLE)
- Added code to automatically update financial_order_state upon an order-state-change notification from Google
- Added code to restore inventory when an order is cancelled
- Refactored code for better organization and clearer logic

To do tomorrow:
- Create “transactions” page in store-admin section that displays past transactions and allows admin to modify or delete a transaction, and allows admin to add a transaction

3/16
Things I did today:
- Added record_restock.php page in store_admin section that allows admin to record restocks in a restocks table and update the inventory accordingly.
- Added an edit_transactions.php page in store_admin section that allows admin to edit transactions. This edits the entries in transactions, transaction_items, and inventory tables.

To do later:
- Modify edit_transactions.php to allow admin to add new transactions (perhaps by specifying “new” in google_order_no

3/20
Things I did today:
- Created add_transactions.php page in store_admin section that allows admin to create transactions that occur outside google checkout. The page allows the admin to choose whether to update inventory based on that transaction.

To do later (next steps):
- Design
- Blog
- Security check
- Start hosting live!
- USPS shipping label printing!
3/28
Things I did today:

- Modified shop page
- Started hosting on remote server for testing
- Created individual product pages with thumbnails
- Researched blog solution; decided to install wordpress in Brotankery.com/blog (later, since we can only have 2 databases at a time)

3/30
Things I did today:

- Researched security and SEO issues
- Moved connect_to_mysql.php file outside root directory
- Added Google analytics code to site
- Wrote a .htaccess file that doesn’t yet work...
- Created robots.txt file to tell crawler not to index WP files or comments or stats

To do later:

- Update Google Checkout info (should anything be different?) - DONE
- Update sandbox links in website code - DONE
- Back up wordpress site, delete files and move my website into public_html - DONE
- Tinker with .htaccess to hide .php extensions, rewrite product.php?pid=0, block direct access to .php files (for duplicate content) - LATER
- Edit Google analytics settings (change domain on “New website” profile) - DONE
- Install wordpress in blog directory - DONE
- Rewrite old blogs
- Add keywords and metadata to home page and shop page - LATER
- Create XML sitemap - DONE
- Improve speed of site, especially shop page

4/1 and 4/2
Things I did today:

- Uploaded code to public_html directory of server – went LIVE
- Updated Google Analytics
- Tested website
- Created blacklight feature in SHOP page
- Decided that .php extensions are fine
- Ran Google PageSpeed on website to optimize load times
- Set images and CSS files to have a 2-day cache timeout
- Wrote first blog entry
- Got photos of new tanks, photoshopped images, and uploaded
- Fixed minor bugs (broken links, incorrect item descriptions)

For the future:

- Generate invoices in admin section
- Generate printing labels in admin section
• Analyze transactions in admin section (sales since date X, tax since date X, number of color/size tanks sold since date X)