Webstore of Wonders Technical Document

To visit the site head here:

http://users.metropolia.fi/~mikkompe/wow/index.php

It is recommended to use a form of incognito mode to ensure no cookie related conflicts appear.

Gain access with these accounts:

userID:	11
accessLevel:	1 (admin)
username:	owner
password:	iDrinkWhisky

userID:	12
accessLevel:	3 (employee)
username:	worker
password:	iDrinkWine

userID:	13
accessLevel:	5 (customer)
username:	shopper
password:	iDrinkBeer

Please see the “flowchart” of the websites workings at:

http://users.metropolia.fi/~mikkompe/wow_flowchart.html

The source files and charts can be found in the wow_project.zip file.

File and class explanations

index.php	Redirects the visitor to loginPage.php.
loginPage.php	prompts the visitor to login to the store.
logoutPage.php	displayed when the user logs out successfully.
purchaseOKPage.php	displayed when the user makes a successful purchase.
storeCustomer.php	normal store page displayed to users with access level 5. Allows for viewing the inventory and adding products to the cart.
storeEmployee.php	same as storeCustomer.php, but adds a link to the administrative page to access level 3 users.
storeAdmin.php	same as storeEmployee.php, but adds the option to view existing users in the database to access level 1 users.
storeEmployeeEdit.php	allows for access level 3 users to edit product quantities and descriptions. Also allows them to change their own password.
storeAdminEdit.php	allows for access level 1 users to add, delete, update product quantity and description, add and remove users, change user access levels and passwords.
cartPage.php	displays the shopping cart's contents and allows for removal of items and purchasing said items in the cart. Same for all access levels.
call_payForStuff.php	calls Store::payForStuff();
call_drawOrderedTable()	calls Store::drawOrderedTable();
dbError.php	displayed if a faulty database query is made.
errorPage.php	generic error page.

config.ini	located outside of public_html. Contains database credentials used by .php files.

Classes

Database.php

The Database class handles connections to the database, querying and possible errors.

dbConnect()	connects to the database using the login credentials found in config.ini, which is located outside of the public_html directory.
dbQuery()	completes a query operation on the database based on the given input
dbSelect()	uses dbQuery() to get some result from the database and stores the returned value to an array, which it returns.
dbError()	displays an error message were something to go wrong with any database related actions.

Login.php

Login handles anything and everything related to logging in and out of the system.

checkCredentials()	connects to the database and gets the row of the username input. If the input password corresponds to the one in the database, the check succeeds. Security is handled with the password_hash() and password_verify() functions provided by PHP. A set amount of “money” is given depending on the access level of the user and control is transferred to profileDirector(). If the check fails, dbError.php is displayed.
profileDirector()	the method directs the user to the

	appropriate store page (admin, employee or customer).
checkLoginStatus()	checks the current username and access level of the user that are stored in $_SESSION. If both of these variables are set and satisfactory, the session name is set. If the current username is not the one required for the page, the user is redirected to index.php. This method is used at the beginning of every file and prevents users from trying to manually enter a higher access level page by entering its URL in the browser.
displayLoginForm()	displays the login form found on loginPage.php.
logout()	empties the shopping cart and calls the Store::sessionLogout() method. A message is displayed of successful logging out and a link is provided to return to index.php.

SessionManager.php

SessionManager takes care of upkeeping killing current sessions.

sessionStart()	starts a session (if one already exists, it is "refreshed")
checkTimeout()	when a user enters a page a new "timer" is set to the current time. If any page is inactive for 5 minutes, the user is logged out and directed to logoutPage.php.
sessionLogout()	handles logging out by unsetting session variables by calling session_unset() and session_destroy(). Returns true if successful;

Store.php

The Store class handles managing the store’s inventory, user base, shopping cart and purchasing, and is responsible for drawing tables on the site.

addProduct()	adds a new product to the store by providing its id, name, category, quantity, price and a description. Will not work with id numbers or product names already in use or if the id number is smaller than one.
removeProduct()	removes a product from the store’s inventory with the provided product id.
updateProductQuantity()	updates a products quantity in the store. Used for ordering more of a certain item or removing units of an item (buying, transferring somewhere else etc). Usable by
updateProductPrice()	sets a new price for a product. May be positive or negative (negative gives the user money if bought).
updateProductDescription()	update a products description.

addUser()	add a user to the database. Provide an id, access level, name and password. The id and username must be unique.
deleteUser()	deletes a user from the database based on the id number given. The admin account cannot be deleted.
updateUserAccessLevel()	change the access level of a user by providing an id number and the new access level. The admin’s access level cannot be changed.
updateUserPassword()	change a user’s password by providing an id number and the new password. The password is hashed and saved to the database. The user admin’s password may only be changed from that account.
updateOwnPassword()	used by accounts with access level 3 (employee) to change their own password.

addToCart()	adds some quantity of a certain product to the shopping cart with the provided id number and quantity. The corresponding

	data is then copied from the products table to the shoppingCart table. Cannot add more products to cart than is available in the store’s inventory.
removeFromCart()	removes an item from the shopping cart. Requires the id number of the product.
payForStuff()	used to complete a purchase transaction. If the cart is empty, a message is displayed informing the shopper. The total price of all items in the cart is then calculated and if the user has enough money, the transaction is completed. If not, a message will be displayed. In the event of a successful transaction, the corresponding number of items is deleted from the products table and the remaining amount of money is displayed. The shoppingCart table is then truncated.

drawOrderedTable()	displays a table of available inventory on the store page ordered according to a parameter acquired from a dropdown menu.
drawTableUsers()	displays a table of all user id numbers, usernames and their access levels in the store ordered by id.
drawTableCart()	displays a table of the current shoppingCart ordered by id.