



Project #4: Warehouse Management System

Team: Can You C My Screen?

CSCI 1110-01: Intro to C Programming

Dr. Reza Sadeghi

Thursday, April 28th, 2021

Outline

Team Description.....	3
System Description.....	5
Functions Description.....	7
Variable Descriptions.....	14
Figures.....	24
References.....	28
Thank You.....	29

Team Description

Arely J. Parra López - Allowing the user to view, request, and save items from the Warehouse Management System as well as providing the admin and guest user an exit function.

Alexander Vita - Adding, deleting, and editing items with varied details (i.e., Type, Stored Time, ID, etc) from Warehouse Management System.

Kamryn Hammond - Creating login page for admin and guest users to enter username & password for Warehouse Management System.

Team Description (cont.)

Jarred Crystal - Generating user-friendly software that provides a welcome page, a menu of all functions that users have access to, and tabular format of all requested information from the Warehouse Management System.

Sophie Ross - Allowing admin to view the list of borrowing requests as well as accept/reject borrowing requests made by guest users in the Warehouse Management System.

System Description

- ◎ **Greeted by a WMS Welcome Page with registration or login.**
- ◎ **If logging in as an Admin, individual is able to:**
 - View a list of usernames & passwords
 - Search for usernames & passwords in the system
 - Change usernames/passwords for admin & guest
 - Add/Remove admin & guest
 - Add/Remove/Edit items from the WMS
 - View/Search for items from the WMS
 - View list of borrowing requests from guest users
 - Accept/Deny borrowing requests from guest users

System Description (cont.)

- ◎ **If logging in as Guest User, individual is able to:**
 - Search for items in the WMS
 - Make a list of favorite items from the WMS
 - Request items to borrow/buy
 - View item's borrowing history
- ◎ **All users can exit from the WMS when in the Login Menu, Admin Menu, or the Guest Menu.**

Functions Description

Crystal:

- ◎ **int loginMenu(void)**
 - Allows for new registration of users and the selection of either the admin or guest login prompt
- ◎ **void main(void)**
 - Provides the main interface to the login menu dictating which function will be executed following user input
- ◎ **int adminMenu(void)**
 - Provides the menu options within the admin menu
- ◎ **void admin(void)**
 - Provides the main interface to the admin menu dictating which function will be executed following user input

Functions Description (cont.)

Crystal:

- ◎ **int guestMenu(void)**
 - Provides the menu options within the guest menu
- ◎ **void guestMain(void)**
 - Provides the main interface to the guest menu dictating which function will be executed following user input

Functions Description (cont.)

Hammond:

- ◎ **int* adminLogin(void)**
 - Allows admin to search for their login from the WMS inventory.
- ◎ **int* guestLogin(void)**
 - Allows guest users to search for their login from the WMS inventory using their private encryption key.
- ◎ **void view(void)**
 - Provides an organized view of all the Admin & Guest User's encryption keys, usernames, and passwords.
- ◎ **int* search(void)**
 - Allows the Admin to search the WMS LogIn.txt file for a specific encryption key, username, and password.

Functions Description (cont.)

Hammond:

- ◎ **void edit(void)**
 - Allows the Admin to select a specific user from the WMS and edit their login information.
- ◎ **void add(void)**
 - Allows the Admin to add a new user to the WMS with an encryption key, username, and password.
- ◎ **void delete(void)**
 - Allows the Admin to select and delete a user from the WMS.

Functions Description (cont.)

Vita:

- ◎ **void addItems (void)**
 - Admin can add an item to the inventory
- ◎ **int* searchItems(void)**
 - Admin can search for an item within the inventory
- ◎ **void editItems (void)**
 - Admin can edit an items characteristics
- ◎ **void deleteItems(void)**
 - Admin can delete an item from the inventory
- ◎ **void viewItems(void)**
 - Admin can view all items within the inventory

Functions Description (cont.)

Ross:

- ◎ **Void acceptDeny(void)**
 - Allows Admin to accept or deny borrowing request list from guest users.
- ◎ **Void viewBorrowing(void)**
 - Allows Admin to view all borrowing requests from guest users.

Functions Description (cont.)

Parra López:

- ◎ **void registration(void)**
 - New individuals can create a new accounts within the WMS.
- ◎ **int* searchWMS(void)**
 - Guest can search for an item from the WMS inventory.
- ◎ **void makeList(void)**
 - Guest can make a list of favorite items from the WMS inventory.
- ◎ **void viewList(void)**
 - Guest can view their list of favorite items.
- ◎ **void requestItems(void)**
 - Guest can request to borrow/buy an item.
- ◎ **void viewHistory(void)**
 - Guest can view the borrowing/purchase history of items.

Variables Description

Crystal:

◎ **int adminMenu(void)**

- int input;

◎ **void admin(void)**

- int option;
- char quit;

◎ **int loginMenu(void)**

- int input;

◎ **int guestMenu(void)**

- int input;

◎ **void guestMain(void)**

- int task;
- char quit;

◎ **void main(void)**

- int choice
- char quit;

Variables Description (cont.)

Hammond:

◎ **int* adminLogin(void)**

◎ **int* guestLogin(void)**

◎ **int* search(void)**

- static int output[2]
- char key[100]
- char input[100]
- FILE *myfile
- char myString[100]
- int line=0
- int find=0
- int fieldNumber=0
- char quit

◎ **void view(void)**

- struct entity
- char encrypt[20]
- char username[20]
- char password[20]
- struct entity new
- char myString[100]
- int pline=0

Variables Description (cont.)

Hammond:

© **void edit(void)**

- int* place=search()
- int line=place[1]
- int find=place[0]
- char username[20], password[20]
- FILE *infile
- FILE *temp
- char myString[100]
- int pline=0
- char quit

© **void add(void)**

- struct entity
- char encrypt[20]
- char username[20]
- char password[20]
- struct entity new
- FILE *myfile
- char quit

Variables Description (cont.)

Hammond:

- ◎ **void delete(void)**
 - int* place=search()
 - int line=place[0]
 - int find=place[0]
 - FILE *originalfile
 - FILE *tempfile
 - char myString[100]
 - int pline=0
 - char quit

Variables Description (cont.)

Vita:

- ◎ **void addItem(void)**
 - struct entity
 - char Type...[100]
 - struct entity new
 - FILE *infile
 - char quit

- ◎ **int* searchItems(void)**
 - static int output[2]
 - char key[100]
 - char input[100]
 - FILE *infile
 - char myString[100]
 - int line=0
 - int find=0
 - int fieldNumber=0
 - int pline=0

Variables Description (cont.)

Vita:

◎ void editItems (void)

- int* place=searchItems()
- int find = 0
- int line = 0
- char Type....[100]
- FILE *invfile
- FILE *tempfile
- char myString[100]
- int pline=0
- char quit

◎ void deleteItems(void)

- int* place=searchItems()
- int find=place[0]
- int line=place[1]
- FILE *invfile
- FILE *tempfile
- char myString[100]
- int pline=0
- char quit

Variables Description (cont.)

Vita:

- ◎ **viewItems(void)**
 - FILE *invfile
 - struct entity
 - char Type....[100;
 - struct entity new
 - char Type....[100];
 - char myString[100000]
 - int pline=0

Variables Description (cont.)

Ross

- **void viewBorrowing(void)**
 - FILE myfile*
 - struct entity
 - char Type...[100]
 - struct entity new
 - char myString[10000]
 - int pline = 0

- **void acceptDeny(void)**
 - struct entity
 - char ID[20]
 - char accept[20]
 - char deny[20]
 - FILE myfile*
 - char quit

Variables Description (cont.)

Parra López:

◎ **void registration(void)**

- struct entity
- char encrypt[20]
- char username[20]
- char password[20]
- struct entity new
- FILE *myfile
- char quit

◎ **int* searchWMS(void)**

- static int output[2]
- char ID[100]
- char input[100]
- FILE *myfile
- char myString[100]
- int line=0
- int find=0
- int fieldNumber=0
- char quit

Variables Description (cont.)

Parra López:

- ◎ **void viewList(void)**
- ◎ **void viewHistory(void)**
 - FILE *myfile
 - struct entity
 - char Type...[100]
 - struct entity new
 - char myString[10000]
 - int pline=0

- ◎ **void makeList(void)**
- ◎ **void requestList(void)**
 - struct entity
 - char Type...[100]
 - char Length[100]
 - struct entity new
 - FILE *myfile
 - char quit

Figures

```
*****
Welcome to the Warehouse Management System (WMS)
*****

*****
Main Log-In Menu
*****

[1] Registration Page
[2] Admin Log-In Page
[3] Guest User Log-In Page
[4] Exit Program

Please select one of the above options to proceed:
```

Figure 7: WMS Main Log-in Menu in Command Prompt

```
*****
Welcome to the Warehouse Management System (WMS)
*****

*****
Admin Menu
*****

[1] View Admin/Guest Usernames and Passwords
[2] Search for Admin/Guest Usernames and/or Passwords
[3] Change the Admin/Guest Usernames and/or Passwords
[4] Add Guest User to WMS
[5] Remove Guest User from WMS
[6] Add Items to Warehouse
[7] Search Items from Warehouse
[8] Edit Items from Warehouse
[9] Delete Items from Warehouse
[10] View Items from Warehouse
[11] View List of Borrowing Requests
[12] Accept or Deny Borrowing Requests
[13] Exit

Please select one of the above options to proceed: _
```

Figure 8: WMS Admin Menu in Command Prompt

Figures (cont.)

```
*****
View Admin/Guest Usernames and Passwords to Log In
*****

Recorded Usernames & Passwords

=====
Key                Username          Password
=====
Arely              aparr3           1234
Alex               avita3           5678
Kamyrn             khamm1           9876
Jarred             jcrys1           5432
Sophie             sross7           1012
reza               reza             reza

Would you like to do another action?(y/n) _
```

Figure 9: WMS Admin/Guest User Encryption Key, Username, & Password Table in Command Prompt

```
*****
View WMS Item Inventory
*****

WMS Item Inventory

=====
ID    Type   Name          Picked          Stored          Provider        Quantity        Place          Price
=====
1234  Book   Stewart_Little 05/18/19_21:25  05/15/19_05:55  B&N             25             Shelf_5        $10.00
5678  Food   Fiji_Apple     05/24/19_20:25  11/15/15_06:35  Dole             15             Shelf_7        $15.00
9876  Book   Hello_Again    08/05/20_16:57  01/01/20_10:41  B&N             31             Shelf_1        $5.00

Would you like to do another action?(y/n) _
```

Figure 10: WMS Inventory Table in Command Prompt

```
*****
View Guest User's Borrowing Requests
*****
```

Guest User's Request List							
Item ID	Item Type	Item Name	Item Provider	Item Quantity	Item Place	Item Price	Borrowing Time
1234	Book	Stewart_Little	B&N	1	Shelf_5	\$10.00	2_weeks

Would you like to do another action?(y/n) _

Figure 11: WMS Guest User's Borrowing Requests Table in Command Prompt

Figures (cont.)

```
*****
Welcome to the Warehouse Managment System (WMS)
*****

*****
Guest Menu
*****

[1] Search WMS for Items in Warehouse
[2] Generate a List of Favorite Items
[3] View List of Favorite Items
[4] Request Borrowing/Buying Items
[5] View Item's Borrowing History
[6] Exit

Please select one of the above options to proceed:
```

Figure 12: WMS Guest User Menu in Command Prompt

Figures (cont.)

```
*****
View Saved List of Favorite Items
*****

Favorite Item List

=====
Item ID  Item Type  Item Name  Item Provider  Item Quantity  Item Place  Item Price
=====
1234    Book      Stewart_Little  B&N           25             Shelf_5     $10.00
5678    Food      Fiji_Apple     Dole           15             Shelf_7     $15.00
9876    Book      Hello_Again    B&N           31             Shelf_1     $5.00

Would you like to do another action?(y/n)
```

Figure 13: WMS Guest User's Favorite Items List Table in Command Prompt

```
*****
History of Borrowed Items
*****

Item History

=====
Item ID  Item Name  Borrower  Taken Out  Returned
=====
1234    Book      Arely_Parra  04/01/2021  04/20/2021
5678    Food      Sophie_Ross  03/28/2021  04/01/2021

Would you like to do another action?(y/n)
```

Figure 14: WMS Guest User's View of Item History Table in Command Prompt

References

1. Fischer, A. E., Fischer, M. J., & Eggert, D. W. (2016, August 30). *Applied Introductory C Programming - University of New Haven*. Chapters from 2nd edition. <http://eliza.newhaven.edu/apc/>.
2. Sadeghi, R. (2020, December 14). *RezaSadeghiWSU/LMS*. GitHub. <https://github.com/RezaSadeghiWSU/LMS>.
3. Sadeghi, R. (2020, December 15). *RezaSadeghiWSU/Calendar-Management-System*. GitHub. <https://github.com/RezaSadeghiWSU/Calendar-Management-System>.
4. Sadeghi, R. (2021). *passwordSystem.c*. West Haven; Reza Sadeghi.

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines. The nodes are represented by small circles, some of which are larger and have concentric circles, suggesting a hierarchical or multi-layered structure. The lines are thin and gray, connecting the nodes in a non-linear fashion.

Thank You