

Assignment #2

Worth: 15% of final grade

Account Ticketing System

Milestone	Worth	Due Date	Submission Required
1	10%	(Suggested Target: March 23 rd)	NO
2	40%	March 26 th by 23:59 EST (end of Week 10)	YES
3	10%	(Suggested Target: April 2 nd)	NO
4	40%	April 9 th by 23:59 EST (end of Week 12)	YES

Introduction

Assignment 2 is a continuation of Assignment 1 Milestone 4 and will complete the account ticketing system. The ticketing component will be added among other improvements.

Additional business and data validation rules will be required to ensure a higher quality data store is maintained including security and overall application functionality.

Customers will be able to login using their credentials and have their own menu of options which include viewing their account information, creating a new ticket, viewing their active tickets, and managing their tickets.

Agents will have more features available from respective main menu options including viewing new tickets, viewing active tickets, managing tickets, and archiving closed tickets.

Persistent storage of data will be implemented so data can be stored (to files) when it changes and reloaded when the application is later restarted. An archiving feature will permit agents to move closed tickets to an archive data file which will help contribute towards better performance and limit resource usage by keeping the active ticket data set lean.

In this assignment, you will be provided with a set of generalized instructions and given **more freedom to create your own solution** (this means creating your own functions, macro's, and deciding in what file they should be placed). **However, you must use the data types and functions that are explicitly stated**).

Preparation

Download or clone the Assignment 2 (A2) from <https://github.com/Seneca-144100/BTP-Project>
In the directory: A2/MS1 you will find the Visual Studio project files ready to load. Open the project (**a2ms1.vcxproj**) in Visual Studio.

IMPORTANT

- ***You will need to copy your work from Assignment 1 Milestone 4 and add all the header and source code files to the a2ms1 Visual Studio project before continuing.***
- ***Remember to update the comments at the top of each file to reflect Assignment 2 Milestone 1.***
- ***DO NOT ADD the a1ms4.c file.***

Milestone – 1 (Worth 10%, Target Due Date: March 23rd)

Milestone – 1 does not require a submission and does not have a specific deadline, however, you should target to have this part completed no later than **March 23rd** to ensure you leave enough time to complete Milestone – 2 which must be submitted and is due **March 26th**.

Milestone-1 includes the main function and should not be modified (***a2ms1.c***). This main will do some preliminary data type testing by creating some test data using the ***new data types*** and then launch the application logic by calling the ***applicationStart*** function accordingly.

This milestone focuses on accommodating some ***new data types*** and extending/improving on data validation routines including implementing additional business logic (rules and conditions for data).

Specifications

New Data Types

You will need to create three (3) new data types in this milestone ("**Message**", "**Ticket**", and "**AccountTicketingData**") which will complete what is needed for this application.

Review the ***a2ms1.c*** file (more specifically the "***main***" and "***populateTickets***" functions) to learn more about the field information used in these new data types based on the data being assigned.

The "**Message**" and "**Ticket**" data types will need to be defined in a **new header file** "***ticket.h***" (don't forget to apply the ***safeguarding*** technique as described in Assignment 1).

Message type

- The **Message** data type has three members. The 1st member is a single character representing the account type of the author of the message (same as used in the **Account** type). The 2nd member represents the display name for a given user (same as used in the **Account** type) and should be able to accommodate 30 printable characters. The 3rd member represents the message details and should be sized to accommodate 150 printable characters.
- Create meaningful member names for each.

Ticket type

- The **Ticket** type has six (6) members.
 1. Unique number for a ticket.
 2. Customer account number related to the ticket.
 3. A ticket status indicator where 0 represents closed and 1 represent active (still open)
 4. A subject line (like an email subject line) that should be able to accommodate up to 30 printable characters.
 5. A counter that represents the number of messages associated with the ticket.
 6. An array of **Message** types that should be able to store up to 20 messages.
- Create meaningful member names for each.

AccountTicketingData type

- This new data type is provided for you below and should be placed in the existing header file "**accountTicketingUI.h**". This type will be used to help simplify the passing of data between key functions with more efficiency and readability.

```
struct AccountTicketingData
{
    struct Account* accounts;    // array of accounts
    const int ACCOUNT_MAX_SIZE; // maximum elements for account array

    struct Ticket* tickets;      // array of tickets
    const int TICKET_MAX_SIZE;   // maximum elements for ticket array
};
```

- Review the **a2ms1.c** file to see how this is instantiated and used.

Application Logic Entry-Point

The function parameters for "**applicationStart**" will need to be modified so it receives just one argument which is a pointer to the new "**AccountTicketingData**" type.

- Update the necessary function definition to use the new argument accordingly (after reading and implementing the menu changes below).

Menu Modifications

menuAgent

- The function used for the **agent** main menu "**menuAgent**" currently has three (3) parameters but will require modification. Since the accounts array and its maximum size information are now members of the new data type "**AccountTicketingData**", the existing first two parameters can be replaced with a pointer to the new "**AccountTicketingData**" type. This means this function should now only have two (2) parameters.
- Update the necessary function definition to use the new argument accordingly.
- The **agentMenu** function should be modified to display four (4) more menu options (6 – 9). Selecting any of these new options will display a temporary notice that the feature is not currently available. Review the sample output for details.

New Client Main Menu

- A new menu needs to be created that will be the main menu for a customer login.
- Currently, when a customer log's in with a customer account number, the application states "CUSTOMER: home menu currently unavailable...". This should be removed and replaced with the customer main menu.
- Review the sample output for the available customer menu options.
- You will need to handle option 1 that displays the account detail information however the other options should display a temporary notice that the feature is not currently available. Review the sample output for details.
- Something to consider: The customer main menu should only have access to Ticket information and the customer's own account record – it should not have access to the system's accounts array information.

Data Validation and Business Rules

You should be applying system library functions like the character analysis and manipulators you have recently learned about to help enforce data validation and business rules where appropriate (review your code and apply where necessary).

New Account

- The application currently prompts the user for an account number when creating a new **Account**. This is not ideal and needs to be replaced with an **auto-generated account number** based on the next increment of the highest number found in the accounts data set.
 - Upgrade your process for creating a new account so the account number is automatically assigned before getting user input for the remaining data. The account number should also be **displayed as part of the title/banner** (see below sample, the **50600** was automatically assigned).
 - Prompting for a new account should therefore start with the account type like this:

```
New Account Data (Account#:50600)
-----
Enter the account type (A=Agent | C=Customer):
```

User Login

- Enhance the validation for obtaining the **UserLogin** member that stores the **login identifier** and do not allow any **whitespace characters** (spaces and tabs etc.). Below is an example of an attempt to enter whitespace characters:

```
User Login Data Input
-----
Enter user login (10 chars max): my login
ERROR: The user login must NOT contain whitespace characters.
Enter user login (10 chars max): my login
ERROR: The user login must NOT contain whitespace characters.
Enter user login (10 chars max):
```

- Enhance the validation for obtaining the **UserLogin** member that stores the **password** to enforce the password meets the new criteria (see example below):

```
Enter the password (must be 8 chars in length): password
SECURITY: Password must contain 2 of each:
    Digit: 0-9
    UPPERCASE character
    lowercase character
    symbol character: !@#$%^&*
Enter the password (must be 8 chars in length): aaAA#$12
```

Note: "aaAA##\$12" is valid because it meets the password validation criteria.

Demographic

- Enhance the **Demographic** process so that entered values for the **country** member are stored as all UPPERCASE characters (the user should be able to enter lowercase characters and you will convert it to uppercase accordingly).

A2-MS1: Sample Output

```
=====
Account Ticketing System - Login
=====
1) Login to the system
0) Exit application
-----

Selection: 1

Enter your account#: 50008

AGENT: Will Smith (50008)
=====
Account Ticketing System - Agent Menu
=====
1) Add a new account
2) Modify an existing account
3) Remove an account
4) List accounts: summary view
5) List accounts: detailed view
-----
6) List new tickets
7) List active tickets
8) Manage a ticket
9) Archive closed tickets
-----
0) Logout

Selection: 1

New Account Data (Account#:50600)
-----
Enter the account type (A=Agent | C=Customer): C

User Login Data Input
-----
Enter user login (10 chars max): Has Space
ERROR: The user login must NOT contain whitespace characters.
Enter user login (10 chars max): NoSpace
Enter the display name (30 chars max): Customer Chris
Enter the password (must be 8 chars in length): A
ERROR: String length must be exactly 8 chars: 12345678
SECURITY: Password must contain 2 of each:
    Digit: 0-9
    UPPERCASE character
    lowercase character
    symbol character: !@#$%^&*
Enter the password (must be 8 chars in length): pa55WD!d
SECURITY: Password must contain 2 of each:
    Digit: 0-9
```

```

        UPPERCASE character
        lowercase character
        symbol character: !@#$%^&*
Enter the password (must be 8 chars in length): pa55WD&!

Demographic Data Input
-----
Enter birth year (current age must be between 18 and 110): 1999
Enter the household Income: $240750.11
Enter the country (30 chars max.): england

*** New account added! ***

```

```
<< ENTER key to Continue... >> [ENTER]
```

```

AGENT: Will Smith (50008)
=====
Account Ticketing System - Agent Menu
=====
1) Add a new account
2) Modify an existing account
3) Remove an account
4) List accounts: summary view
5) List accounts: detailed view
-----
6) List new tickets
7) List active tickets
8) Manage a ticket
9) Archive closed tickets
-----
0) Logout

```

```
Selection: 5
```

Acct#	Acct.Type	Birth	Income	Country	Disp.Name	Login	Password
30001	CUSTOMER	1990	\$ 150000.10	CANADA	Silly Sally	cust1	A*1*3*D*
50599	AGENT	1972	\$2250400.22	AFRICA	Fred Flintstone	agent1	y*b*#*@*
30004	CUSTOMER	1978	\$ 250800.74	INDIA	Betty Boop	cust2	1*E*P*!*
50008	AGENT	1952	\$2350600.82	U.S.A.	Will Smith	agentJ	T***2*t*
20020	CUSTOMER	2000	\$ 350500.35	KOREA	Shrimpy Shrimp	cust3	8*3*J*m*
50600	CUSTOMER	1999	\$ 240750.11	ENGLAND	Customer Chris	NoSpace	p*5*W*&*

```
<< ENTER key to Continue... >> [ENTER]
```

```

AGENT: Will Smith (50008)
=====
Account Ticketing System - Agent Menu
=====
1) Add a new account
2) Modify an existing account
3) Remove an account
4) List accounts: summary view

```

```
5) List accounts: detailed view
-----
6) List new tickets
7) List active tickets
8) Manage a ticket
9) Archive closed tickets
-----
0) Logout

Selection: 2

Enter the account#: 50600

Account: 50600 - Update Options
-----
1) Update account type (current value: C)
2) Login
3) Demographics
0) Done
Selection: 2

User Login: NoSpace - Update Options
-----
1) Display name (current value: Customer Chris)
2) Password
0) Done
Selection: 2

Enter the password (must be 8 chars in length): juMP1!*&
SECURITY: Password must contain 2 of each:
    Digit: 0-9
    UPPERCASE character
    lowercase character
    symbol character: !@#$$%^&*
Enter the password (must be 8 chars in length): juMP1!*9

User Login: NoSpace - Update Options
-----
1) Display name (current value: Customer Chris)
2) Password
0) Done
Selection: 0

Account: 50600 - Update Options
-----
1) Update account type (current value: C)
2) Login
3) Demographics
0) Done
Selection: 3

Demographic Update Options
-----
```

```
1) Household Income (current value: $240750.11)
2) Country (current value: ENGLAND)
0) Done
Selection: 2
```

```
Enter the country (30 chars max.): romania
```

```
Demographic Update Options
```

```
-----
1) Household Income (current value: $240750.11)
2) Country (current value: ROMANIA)
0) Done
Selection: 0
```

```
Account: 50600 - Update Options
```

```
-----
1) Update account type (current value: C)
2) Login
3) Demographics
0) Done
Selection: 0
```

```
AGENT: Will Smith (50008)
```

```
=====
Account Ticketing System - Agent Menu
=====
```

```
1) Add a new account
2) Modify an existing account
3) Remove an account
4) List accounts: summary view
5) List accounts: detailed view
```

```
-----
6) List new tickets
7) List active tickets
8) Manage a ticket
9) Archive closed tickets
```

```
-----
0) Logout
```

```
Selection: 6
```

```
Feature #6 currently unavailable!
```

```
<< ENTER key to Continue... >> [ENTER]
```

```
AGENT: Will Smith (50008)
```

```
=====
Account Ticketing System - Agent Menu
=====
```

```
1) Add a new account
2) Modify an existing account
3) Remove an account
4) List accounts: summary view
```


5) List accounts: detailed view

6) List new tickets

7) List active tickets

8) Manage a ticket

9) Archive closed tickets

0) Logout

Selection: 7

Feature #7 currently unavailable!

<< ENTER key to Continue... >> [ENTER]

AGENT: Will Smith (50008)

Account Ticketing System - Agent Menu

1) Add a new account

2) Modify an existing account

3) Remove an account

4) List accounts: summary view

5) List accounts: detailed view

6) List new tickets

7) List active tickets

8) Manage a ticket

9) Archive closed tickets

0) Logout

Selection: 8

Feature #8 currently unavailable!

<< ENTER key to Continue... >> [ENTER]

AGENT: Will Smith (50008)

Account Ticketing System - Agent Menu

1) Add a new account

2) Modify an existing account

3) Remove an account

4) List accounts: summary view

5) List accounts: detailed view

6) List new tickets

7) List active tickets

8) Manage a ticket

9) Archive closed tickets

0) Logout

Selection: 9

Feature #9 currently unavailable!

<< ENTER key to Continue... >> [ENTER]

AGENT: Will Smith (50008)

=====
Account Ticketing System - Agent Menu
=====

- 1) Add a new account
- 2) Modify an existing account
- 3) Remove an account
- 4) List accounts: summary view
- 5) List accounts: detailed view

-
- 6) List new tickets
 - 7) List active tickets
 - 8) Manage a ticket
 - 9) Archive closed tickets
-

0) Logout

Selection: 0

LOGGED OUT

=====
Account Ticketing System - Login
=====

- 1) Login to the system
 - 0) Exit application
-

Selection: 1

Enter your account#: 12345

ERROR: Access Denied.

<< ENTER key to Continue... >> [ENTER]

=====
Account Ticketing System - Login
=====

- 1) Login to the system
 - 0) Exit application
-

Selection: 1

Enter your account#: 50600

CUSTOMER: Customer Chris (50600)

=====

Customer Main Menu

=====

- 1) View your account detail
- 2) List my tickets
- 3) Create a new ticket
- 4) Manage a ticket

0) Logout

Selection: 1

Acct#	Acct.Type	Birth	Income	Country	Disp.Name	Login	Password
50600	CUSTOMER	1999	\$ 240750.11	ROMANIA	Customer Chris	NoSpace	j*M*1***

<< ENTER key to Continue... >> [ENTER]

CUSTOMER: Customer Chris (50600)

=====

Customer Main Menu

=====

- 1) View your account detail
- 2) List my tickets
- 3) Create a new ticket
- 4) Manage a ticket

0) Logout

Selection: 2

Feature #2 currently unavailable!

<< ENTER key to Continue... >> [ENTER]

CUSTOMER: Customer Chris (50600)

=====

Customer Main Menu

=====

- 1) View your account detail
- 2) List my tickets
- 3) Create a new ticket
- 4) Manage a ticket

0) Logout

Selection: 3

Feature #3 currently unavailable!

```
<< ENTER key to Continue... >> [ENTER]

CUSTOMER: Customer Chris (50600)
=====
Customer Main Menu
=====
1) View your account detail
2) List my tickets
3) Create a new ticket
4) Manage a ticket
-----
0) Logout

Selection: 4

Feature #4 currently unavailable!

<< ENTER key to Continue... >> [ENTER]

CUSTOMER: Customer Chris (50600)
=====
Customer Main Menu
=====
1) View your account detail
2) List my tickets
3) Create a new ticket
4) Manage a ticket
-----
0) Logout

Selection: 0

### LOGGED OUT ###

=====
Account Ticketing System - Login
=====
1) Login to the system
0) Exit application
-----

Selection: 0

Are you sure you want to exit? ([Y]es|[N]o): y

=====
Account Ticketing System - Terminated
=====
```

Milestone – 1 Submission

1. ***This is a test submission for verifying your work only*** – no files will be submitted to your instructor.
2. Upload (file transfer) your all header and source files:
 - **a2ms1.c**
 - **account.c**
 - **account.h**
 - **accountTicketingUI.c**
 - **accountTicketingUI.h**
 - **commonHelpers.c**
 - **commonHelpers.h**
 - **ticket.h**
3. Login to matrix in an SSH terminal and change directory to where you placed your source code.
4. Manually compile and run your program to make sure everything works properly:

```
gcc -Wall a2ms1.c account.c accountTicketingUI.c commonHelpers.c -o  
ms1 <ENTER>
```

*If there are no error/warnings are generated, execute it: **ms1** <ENTER>*

5. Run the submission command below (replace **profname.proflastname** with **your professors** Seneca userid and replace **NAA** with your section):

```
~profName.proflastname/submit 100a2ms1/NAA_ms1 <ENTER>
```

6. Follow the on-screen submission instructions.
-