

Specific expected tasks and examples

In this assignment, you are required to develop a console-based, menu driven hospital management system using C#. Necessary data should be stored in “.txt” files. There are 3 core roles in the system.

1. Patient
2. Doctor
3. Administrator

Each of these roles have their own version of the menu with it's own subset of menu options.

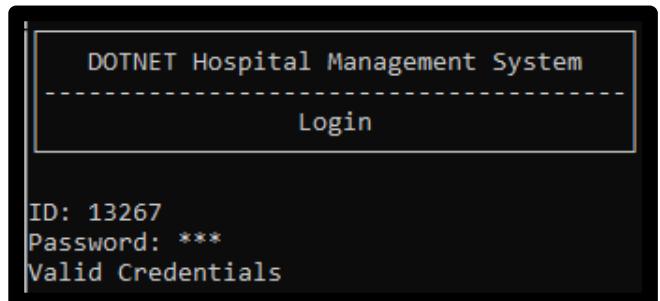
Login Menu

The user should be presented with this menu on program startup and cannot proceed further without entering valid login details.

Input fields:

- ID: Display the typed characters as is
- Password: Display “*” instead of the actual characters

ID and password should be checked against valid credentials inside your files. How you do this depends on how you've structured your classes; but any patient/doctor or admin should have a unique ID and a password. These values should be, in some way, stored in a .txt file, which your code should cross check at this screen. If entered values are not valid, display appropriate error message and allow the user to retry.

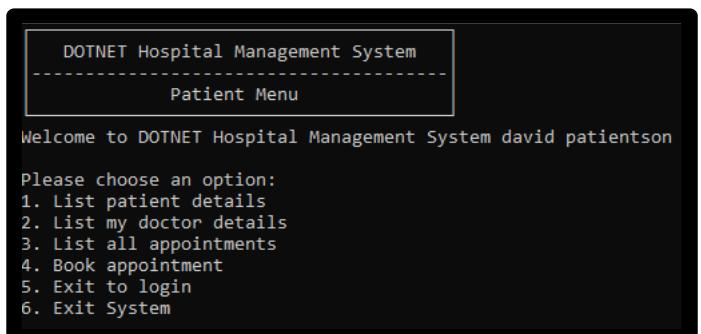


Patient Menu

The patient menu should have all of the following menu options (see right).

All menu options MUST share this functionality (see example below):

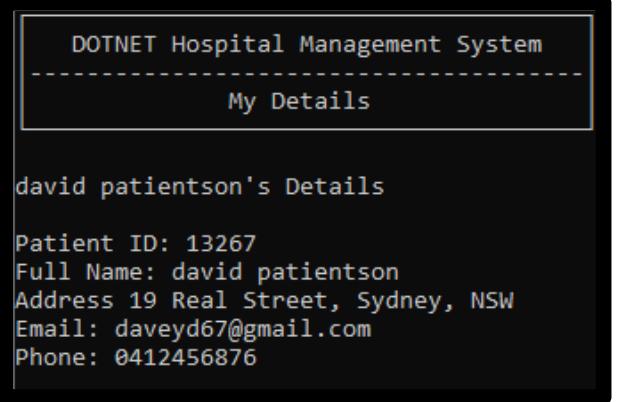
- Console is emptied to display new data
- Appropriate heading displayed before data printed
- Error handling for bad user input
- Menu should never return itself, the user should be able to press a single key to return.



List Patient Details

Functionality: Lists all the fields of the currently logged in patient to the console.

Input: Key press to return to menu.



List My Doctor Details

Functionality: Lists all the fields of the doctor that is registered with the currently logged in patient, to the console. It is recommended that the Doctor data structure has a `toString` function which compresses the necessary details of the patient into a short line.

DOTNET Hospital Management System			
My Doctor			
Your doctor:			
Name	Email Address	Phone	Address
jack doctorson	drjack@gmail.com	0412333676	23 Real Street, Sydney, NSW

List All Appointments

Functionality: Lists the details of all past appointments involving the currently logged in patient.

Input: Key press to return to menu.

DOTNET Hospital Management System		
My Appointments		
Appointments for david patientson		
Doctor	Patient	Description
jack doctorson	david patientson	cold symptoms
jack doctorson	david patientson	regular checkup with doc

Book Appointment

Functionality: Prompts the user for all the necessary information to generate a new appointment. This must be done with the doctor registered to the user. If the user has not registered to a doctor, they should be prompted to choose from a list of all doctors and register.

Input:

- Integer to select doctor (if applicable)
- All necessary information to generate your Doctor data structure
- Key press to return to menu.

DOTNET Hospital Management System	
Book Appointment	
You are not registered with any doctor! Please choose which doctor you would like to register with	
1 david doctorson drdavid@gmail.com 0412456876 19 Real Street, Sydney, NSW	
2 jack doctorson drjack@gmail.com 0412333676 23 Real Street, Sydney, NSW	
Please choose a doctor:	
1	
You are booking a new appointment with david doctorson	
Description of the appointment: got the flu	
The appointment has been booked successfully	
DOTNET Hospital Management System	
Book Appointment	
You are booking a new appointment with david doctorson	
Description of the appointment: flu again	
The appointment has been booked successfully	

Logout

Functionality: Returns to the login menu

Exit

Functionality: Exits the application

Doctor Menu

The doctor menu should have all of the following menu options (see right)

All menu options MUST share this functionality:

- Console is emptied to display new data
- Appropriate heading displayed before data printed
- Error handling for bad user input
- Menu should never return itself, the user should be able to press a single key to return.

```
-----  
| DOTNET Hospital Management System |  
| Doctor Menu |  
-----  
Welcome to DOTNET Hospital Management System jack doctorson  
Please choose an option:  
1. List doctor details  
2. List patients  
3. List appointments  
4. Check particular patient  
5. List appointments with patient  
6. Logout  
7. Exit
```

List Doctor Details

Functionality: Lists the fields of the currently logged in doctor to the console.

Input: Key press to return to menu.

```
-----  
| DOTNET Hospital Management System |  
| My Details |  
-----  
Name | Email Address | Phone | Address  
-----  
jack doctorson | drjack@gmail.com | 0412333676 | 23 Real Street, Sydney, NSW
```

List Patients

Functionality: Lists a shorthand outline of every patient that is registered with the currently logged in doctor to the console, line by line. It is recommended that the Patient data structure has a `toString` function which compresses the necessary details of the patient into a short line.

```
-----  
| DOTNET Hospital Management System |  
| My Patients |  
-----  
Patients assigned to jack doctorson:  
Patient | Doctor | Email Address | Phone | Address  
-----  
Davey Flu | jack doctorson | david@david.com | 0412341234 | 12 Real Place, Sydney, NSW
```

List Appointments

Functionality: Lists every appointment involving the currently logged in doctor, regardless of which patient is involved, to the console, line by line. It is recommended that that the Appointment data structure has a `ToString` function which compresses the necessary details of the appointment into a short line.

Input: Key press to return to menu.

DOTNET Hospital Management System		
All Appointments		
Doctor	Patient	Description
jack doctorson	david patientson	regular checkup with doc
jack doctorson	Davey Flu	covid

Check Particular Patient

Functionality: Prompts the user for an ID and prints the details of the patient whose ID it belongs to, to the console line by line. If there is no patient with that ID, your code should handle the error and print an appropriate error message. It is your choice whether the user is prompted to re-enter the ID or if the program returns to the Patient menu.

Input: Key press to return to menu.

DOTNET Hospital Management System				
Check Patient Details				
Enter the ID of the patient to check: 13267				
Patient	Doctor	Email Address	Phone	Address
david patientson	jack doctorson	daveyd67@gmail.com	0412456876	19 Real Street, Sydney, NSW

List Appointments With Patient

Functionality: Prompts the user for an ID and finds the patient with that ID. Your code should then print every appointment between that patient and the currently logged in doctor. (You should not need to filter this list, as the patient should have been forced to be registered to only 1 doctor). If there is no patient with that ID, your code should handle the error and print an appropriate error message. It is your choice whether the user is prompted to re-enter the ID or if the program returns to the Patient menu.

Input: Key press to return to menu.

DOTNET Hospital Management System		
Appointments With		
Enter the ID of the patient you would like to view appointments for: 83618		
Doctor	Patient	Description
Jack doctorson	Davey Flu	covid

Logout

Functionality: Returns to the login menu

Exit

Functionality: Exits the application

Administrator menu

The administrator menu should have all of the following menu options (see right)
All menu options MUST share this functionality:

- Console is emptied to display new data
- Appropriate heading displayed before data printed
- Error handling for bad user input
- Menu should never return itself, the user should be able to press a single key to return.

```
DOTNET Hospital Management System
-----
Administrator Menu

Welcome to DOTNET Hospital Management System david adminson

Please choose an option:
1. List all doctors
2. Check doctor details
3. List all patients
4. Check patient details
5. Add doctor
6. Add patient
7. Logout
8. Exit
```

List All Doctors

Functionality: Lists a shorthand version of every doctor contained in the system to the console, line by line. It is recommended that the Doctor data structure has a `toString` function which compresses the necessary details of the doctor into a short line.

Input: Key press to return to menu.

```
DOTNET Hospital Management System
-----
All Doctors

All doctors registered to the DOTNET Hospital Management System

Name | Email Address | Phone | Address
-----
david doctorson | drdavid@gmail.com | 0412456876 | 19 Real Street, Sydney, NSW
jack doctorson | drjack@gmail.com | 0412333676 | 23 Real Street, Sydney, NSW
```

Check Doctor Details

Functionality: Prompts the user for an ID and prints the details of the doctor whose ID it belongs to, to the console line by line. If there is no doctor with that ID, your code should handle the error and print an appropriate error message. It is your choice whether the user is prompted to re-enter the ID or if the program returns to the Administrator menu.

Input: Key press to return to menu.

```
DOTNET Hospital Management System
-----
Doctor Details

Please enter the ID of the doctor who's details you are checking. Or press n to return to menu
32423

Details for jack doctorson

Name | Email Address | Phone | Address
-----
jack doctorson | drjack@gmail.com | 0412333676 | 23 Real Street, Sydney, NSW
```

List All Patients

Functionality: Lists the shorthand version of every patient in the system to the console, line by line. It is recommended that the Patient data structure has a `toString` function which compresses the necessary details of the patient into a short line.

Input: Key press to return to menu.

```
DOTNET Hospital Management System
-----
All Patients

All patients registered to the DOTNET Hospital Management System

Patient | Doctor | Email Address | Phone | Address
-----
david patientson | jack doctorson | daveyd67@gmail.com | 0412456876 | 19 Real Street, Sydney, NSW
Davey Flu | jack doctorson | david@david.com | 0412341234 | 12 Real Place, Sydney, NSW
```

Check Patient Details

Functionality: Prompts the user for an ID and prints the details of the doctor whose ID it belongs to, to the console line by line. If there is no doctor with that ID, your code should handle the error and print an appropriate error message. It is your choice whether the user is prompted to re-enter the ID or if the program returns to the Administrator menu.

Input: Key press to return to menu.

```
DOTNET Hospital Management System
-----
Patient Details

Please enter the ID of the patient who's details you are checking. Or press n to return to menu
83618

Details for Davey Flu

Patient | Doctor | Email Address | Phone | Address
-----
Davey Flu | jack doctorson | david@david.com | 0412341234 | 12 Real Place, Sydney, NSW
```

Add Doctor

Functionality: Prompts the user for all the necessary data needed to generate a new Doctor and add it to the system.

Input: Key press to return to menu.

```
DOTNET Hospital Management System
-----
Add Doctor

Registering a new doctor with the DOTNET Hospital Management System
First Name: Dr
Last Name: Avinash
Email: avinash@dnhms.com
Phone: 04123123
Street Number: 45
Street: Real Place
City: Sydney
State: NSW
Dr Avinash added to the system!
```

Add Patient

Functionality: Prompts the user for all the necessary data needed to generate a new Patient and add it to the system.

Input: Key press to return to menu.

```
DOTNET Hospital Management System
-----
Add Patient

Registering a new patient with the DOTNET Hospital Management System
First Name: Davey
Last Name: Flu
Email: verysick@gmail.com
Phone: 04123123
Street Number: 87
Street: Real Place
City: Sydney
State: NSW
Davey Flu added to the system!
```

Logout

Functionality: Returns to the login menu

Exit

Functionality: Exits the application