

COMPSCI 280: Assignment 3

In this assignment, you will add a GUI front-end to the application you developed in Assignment 2. This will give you hands-on experience working with WinForms in *Microsoft Visual Studio*.

This assignment is due on **Friday 21 October, 2016, 11:59pm** and is worth 5% of your final course marks.

Estimated time to complete this assignment is 15-20 hours.

Background

The ambulance service for the city of Chromatropolis is in the process of updating their computer systems. They have reviewed your system from the previous assignment and decided it needs a Graphical User Interface (GUI) and some additional entry and validation.

Your task for this assignment is to extend the application for tracking ambulances officers from assignment 2. This involves changing to a GUI and adding additional functionality.

Submission

To prepare your solution for submission, ensure it is working on the lab computers, then delete the following directories:

- bin
- obj
- packages

This will leave a minimal set of solution files from which the marker can do a rebuild on their machine.

Submit via the Assignment Dropbox (). After deleting the directories as above, create a single .zip file of the entire solution directory and submit that **one** file via the Dropbox. Note:

- Ensure you submit for the correct course.
- If you resubmit, please include **all** files in the .zip file of your re-submission.

Requirements

For this assignment you will be adding a GUI front-end to the **ambulance staff rostering system** you developed in assignment 2. You will also extend the application to add additional functionality.

You will not need to modify the database you developed for assignment 2. However you will need to include the file 'Initialise.sql' from assignment 2 in your submission so the marker can generate your database.

A) Creating the assignment project

1. Create your project with *Microsoft Visual Studio 2015*
2. Ensure 'Create Directory for solution' is checked, set the name and solution name to 'Assignment3' (no spaces, no quotes)
3. Create a Visual C# Windows Form application

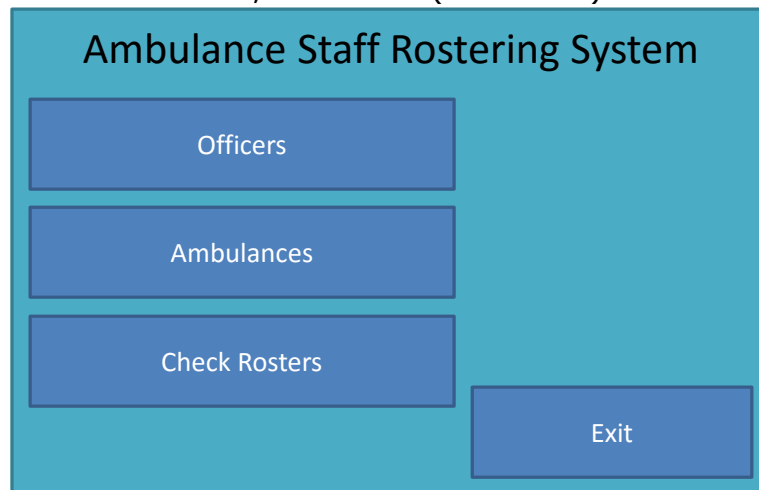
B) Establishing interoperability

1. As per the week 8 lab sheet, set up your solution with packages for EntityFramework and MySQL

C) GUI interface

1. Menu

- This screen is the entry point for the application, it has four buttons: "Officers", "Ambulances", "Check Rosters", and "Exit" (see below)



- Clicking on the "Officers" button will display the **Ambulance Officer List** screen
- Clicking on the "Ambulances" button will display the **Ambulance List** screen
- Clicking on the "Check Rosters" button will display the **Check Rosters** screen
- Clicking on the "Exit" button will end the application

2. Ambulance Officer List

- This screen displays all the officers in the database, it has a list of all the ambulance officers and two buttons: "Back" and "Add New" (see below)

Ambulance Officers			
ID	Name	Skill	Ambulance
135790	John Doe	Basic	A7
135970	Peter Smith	Basic	None
131234	Jane Doe	Intermediate	A41
133535	Bill Bobbins	Intermediate	A110
130001	Carol Green	Advanced	None
132244	Jill Shield	Basic	None
134321	John Quick	Basic	A41

Back Add New

- The list of ambulance officers displays the ID, name (first names plus surname), skill, and assigned ambulance of all the ambulance officers in the database
- Clicking on an ambulance officer will display the **Add/Edit Officer** screen with the clicked officer's details loaded
- Clicking on the "Back" button will display the **Menu** screen
- Clicking on the "Add New" button will display the **Add/Edit Officer** screen with no details in it
- Bonus mark:* implement sorting when the user clicks on a column heading

3. Add/Edit Officer

- This screen allows the user to edit the details of an ambulance officer, it has a label and input field for each database field and two buttons: "Cancel" and "Save" (see below)

Ambulance Officer: John Doe

First Names: John

Surname: Doe

Officer ID: 135970

Skill Level: Basic

Ambulance: A7

Cancel Save

- The first names field allows the user to add/edit the first name of the ambulance officer. This field is required and has a minimum length of one.
- The surname field allows the user to add/edit the surname of the ambulance officer. This field is required and has a minimum length of one.
- The officer ID field allows the user to add/edit the ID of the ambulance officer. This field is required; the data must be a six digit number.
- The skill level allows the user to set the skill level of the officer. It is a dropdown combobox with three values: "Basic", "Intermediate", and "Advanced". The user cannot add new values to this list.
- The ambulance field allows the user to assign or unassign the officer to an ambulance. It is a dropdown combobox. The first (default) value is "None". The remaining values are populated from the Ambulance table in the database. These values must be in alphanumeric order. The user cannot add new values to this list.
- Clicking on the "Back" button will display the **Ambulance Officer List** screen
- Clicking on the "Save" will validate and save the data:
 - ❖ If the data is invalid, a message will be displayed to the user with the details of the invalid data (e.g. "First names is a required field")
 - ❖ If the data is valid, it will update their record in the database. For an existing officer, this will update their details. For a new officer, this will add a new record to the database.
- *Bonus mark:* when validating the data, ensure that the officer ID is unique

4. Ambulance List

- This screen displays all the ambulances in the database, it has a list of all the ambulances and two buttons: "Back" and "Add New" (see below)

The screenshot shows a screen titled "Ambulances". It contains a table with two columns: "ID" and "Station". The table lists three ambulances: A7 at Greenfields, A42 at Bluelane, and A110 at Redville. Below the table are two buttons: "Back" and "Add New".

ID	Station
A7	Greenfields
A42	Bluelane
A110	Redville

- The list of ambulances displays the ID, and station of all the ambulances in the database
- Clicking on an ambulance will display the **Add/Edit Ambulance** screen with the clicked ambulance's details loaded
- Clicking on the "Back" button will display the **Menu** screen
- Clicking on the "Add New" button will display the **Add/Edit Ambulance** screen with no details in it
- Bonus mark:* implement sorting when the user clicks on a column heading

5. Add/Edit Ambulance

- This screen allows the user to edit the details of an ambulance, it has a label and input field for each database field and two buttons: "Cancel" and "Save" (see below)

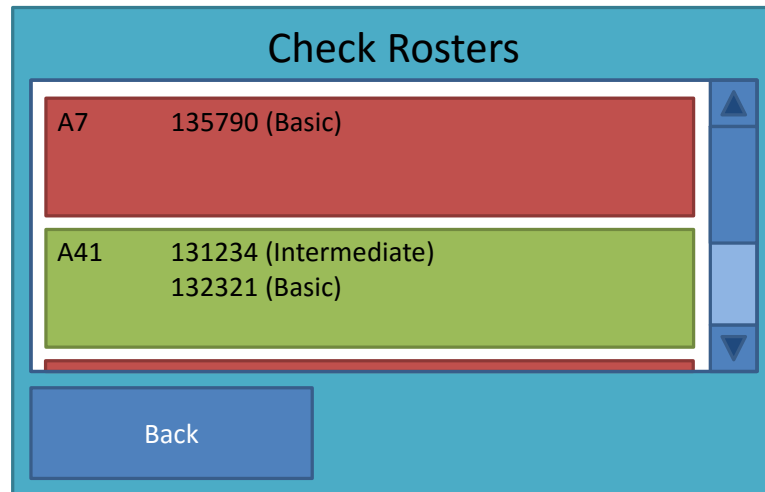
The screenshot shows a screen titled "Ambulance: A7". It contains three input fields: "Ambulance ID:" with the value "A7", "Station:" with the value "Greenfields", and "Crew:" with the value "John Doe". Below the input fields are two buttons: "Cancel" and "Save".

- The ambulance ID field allows the user to add/edit the ID of the ambulance. This field is required; the first character must be an A followed by one or more digits (up to three).
- The station field allows the user to add/edit the station the ambulance is based at. This field is required and has a minimum length of one.
- The crew field is a read-only text box containing the crew members assigned to the ambulance, one per line. The officers' full names are displayed (first names plus surname)
- Clicking on the "Back" button will display the **Ambulance List** screen
- Clicking on the "Save" will validate and save the data:

- ❖ If the data is invalid, a message will be displayed to the user with the details of the invalid data (e.g. "Ambulance ID must consist of an A followed by a number")
- ❖ If the data is valid, it will update their record in the database. For an existing ambulance, this will update their details. For a new ambulance, this will add a new record to the database.
- *Bonus mark:* change the station input field to a combobox and populate with the list of existing stations in the database (the user can still type in a new station)

6. Ambulance List

- This screen displays all the ambulances in the database and whether they have sufficient crew. It has a list of all the ambulances and one button: "Back" (see below)



- The list of ambulances displays a card for each ambulance. Each card displays:
 - ❖ The ID of the ambulance in the top-left corner.
 - ❖ The ID and skill level of each assigned ambulance officer in the centre.
 - ❖ The card will be green if there is sufficient crew for the ambulance, red otherwise (see below).
- Clicking on the "Back" button will display the **Menu** screen

Ambulance Rules

The following rules are used to specify whether an ambulance has sufficient crew:

- An ambulance must have between two and three crew
- At least one crew member must have Intermediate or Advanced skill

Mark Scheme

The marker will be looking for the following:

Item	Points
All screens correctly implemented, with data being populated from the database	5
Clicking on the button takes the user to the correct screen	5
Clicking on save correctly updates the database (ambulance officers and ambulances)	5
Data is correctly validated prior to being saved (ambulance officers and ambulances)	5
The check roster screen correctly displays the ambulances correctly (details and colour)	5
<i>Bonus marks*</i>	4
Total (maximum mark)	25

* The maximum possible mark for this assignment is 25. Bonus marks allow you to compensate for lower marks in other parts of this assignment

Penalties

Marks will be deducted from your total for the following:

Penalty	Description
-5	Error in the application – applies if your program throws a run-time exception from any normal input by the marker
Up to -25	Hard coding – if the marker finds that you’ve inappropriately entered data or case-specific responses directly into your C# code rather than implementing general solutions using the DBMS
??	Late submission – the lecturer may accept late submissions by a scheme of penalties announced only after the deadline has passed. Note that extensions without penalty are possible for medical or compassionate grounds (not including workload in other courses, normal employer requirement or elective travel), or through request of University Counselling Services. The dropbox will remain open for a substantial period after the initial deadline to allow receipt of late submissions.