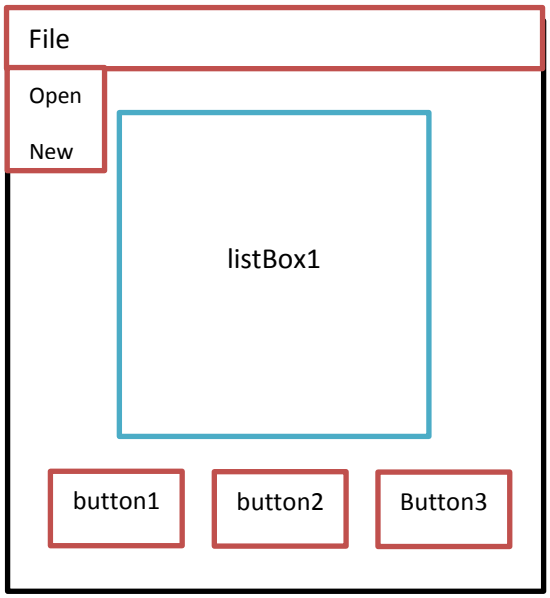
Marc Anthony Palminteri

Task A

1. Produce a suitable diagrammatical representation of the program to show the structure and components.

Form1.cs:



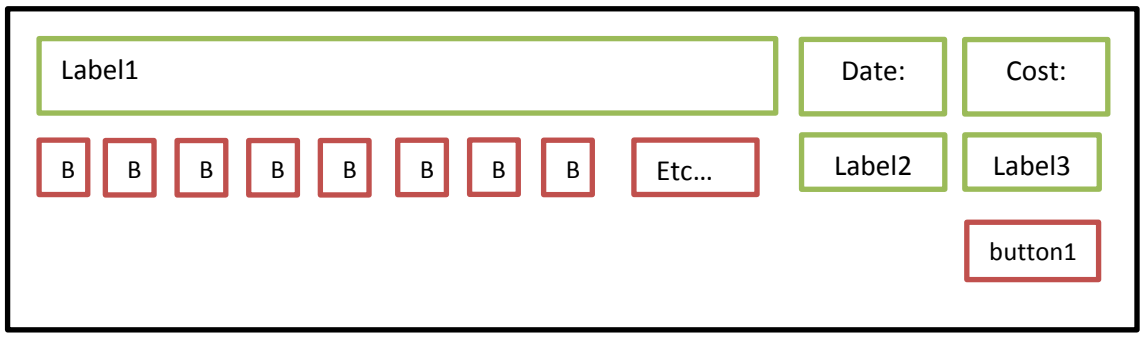
listBox1: The list populates when the user opens a text file via the File > Open menu.

button1: After the user has selected a course from listBox1 it will open the relevant SeatPlan.cs. If a course isn't selected, an error message will occur telling the user to select one.

button2: If the user has opened a text file, they can choose the second button with opens up AddCourse.cs which allows them to add a course.

button3: Refreshes listBox1 with new data

SeatPlan.cs:



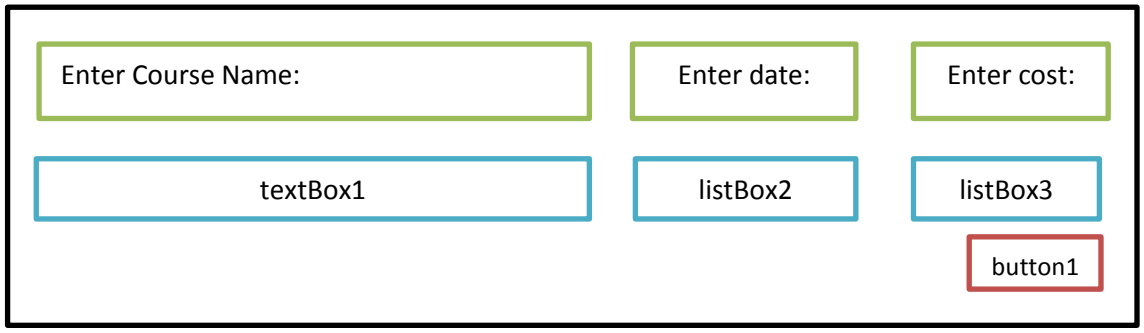
label1: Course name

labe2: Current course Date

label3: Current course cost

button1: Sends the updated information back to form 1 and closes SeatPlan.cs

AddCourse.cs:



textBox1: User inputs the new course name

textBox2: User inputs the date the course starts

textBox3: User inputs the cost per person

button1: Send the information back to form 1 and closes AddCourse.cs

2. Produce the program design language algorithms for the following:

• operation of the Booking Seat Plan screen

When the user toggles a button to check or uncheck a booked place on the course, the CheckChanged method will the change the buttons state, colour and text.

```
private void CheckChanged(object sender, EventArgs e){
      if (check){
            check = false;
            this.Text = this.TabIndex.ToString();
            this.BackColor = System.Drawing.SystemColors.Control;
      }
else{
      check = true;
      this.Text = "B";
      this.BackColor = System.Drawing.Color.Lime;
      }
}
```

• creation of a new file

This will create a new form that requests the input of the new course information for a new file.

```
public void newToolStripMenuItem_Click(object sender, EventArgs e){
            AddCourse newCourse = new AddCourse(text, listBox1, names);
            newCourse.Show();
 }
```

• opening of an existing file.

When the user clicks the File -> Open button, the program will open up a window that asks the user to find the required text file (.txt) to open

```
private void openToolStripMenuItem_Click(object sender, EventArgs e){
      OpenFileDialog myDialog = new OpenFileDialog();
      myDialog.Title = "Open file";
      myDialog.Filter = "TXT files|*.txt";
      myDialog.InitialDirectory = @"THE USERS DESKTOP";
}
```

3. Specify any error handling needed to trap errors in the booking screen, file creation and opening an existing file algorithms.

• operation of the Booking Seat Plan screen

The CheckChanged method checks if the button has already been pressed before using the if/else statement "if (check)". This will then change the button to its other 'state' when clicked.

```
private void CheckChanged(object sender, EventArgs e){
      if (check){
            check = false;
            this.Text = this.TabIndex.ToString();
            this.BackColor = System.Drawing.SystemColors.Control;
      }
      else{
            check = true;
            this.Text = "B";
            this.BackColor = System.Drawing.Color.Lime;
      }
}
```

• creation of a new file

The if/else statement in the 'new file' section now checks that a file has be previously opened. If not, an error will occur.

```
public void newToolStripMenuItem_Click(object sender, EventArgs e){
      if (text != null){
            AddCourse newCourse = new AddCourse(text, listBox1, names);
            newCourse.Show();
      }
      else{
            MessageBox.Show("A file was not opened", "Error",
      MessageBoxButtons.OK, MessageBoxIcon.Warning);
      }
}
```

• opening of an existing file.

The code will now check that something acceptable was selected when the user was asked to open a text file.

```
if (myDialog.ShowDialog() == DialogResult.OK){
      //Do something with the code
}
else {
      MessageBox.Show("Error code 002\nFile open error or dialog
      cancelled", "Error", MessageBoxButtons.OK, MessageBoxIcon.Warning);
}
```

4. Give a brief description of the importance of software design in the Systems Development Life Cycle (SDLC).

The software development life cycle has 5 stages:

Planning - Planning for the quality assurance requirements and identification of the possible problems associated with the project is also done in the planning stage. The outcome of the developers research will define the various approaches to implementing the project successfully with minimum risks.

Analysis - Requirement analysis is a very important and fundamental stage in SDLC. It is done by asking the the customer, the sales department, market surveys and domain experts in the industry for their input. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational, and technical areas.

Design – The design approach clearly defines all the areas of the product along with its communication and data flow representation with the external and third party modules. The internal design of all the modules of the proposed architecture should be clearly defined in its entirety with as much detail as possible.

Testing - The testing activities are mostly involved in all the stages of SDLC but this stage refers to the testing only stage of the product where products defects are reported, tracked, fixed and retested, until the product reaches the quality standards.

Maintenance - Once the product is tested and ready to be deployed, it is released formally in the appropriate market. Sometimes product deployment happens in stages as per the organizations business strategy. The product may first be released in a limited segment and tested in the real business environment. Based on the feedback, the product may be released as it is or with suggested enhancements. After the product is released in the market, its maintenance is done for the existing customer base.