

COMP1004 – Rapid Application Development

Assignment 3

Movie Bonanza – Online Streaming.

Due Week #8 (Sunday March 5, 2016) @midnight.

Value 15%

Movie Bonanza – Online Streaming.

Maximum Mark: 57

Overview: Create a multi-form project that allows the user to select a Movie to view from an online video streaming service called “Movie Bonanza”. The user will select a Movie from a Drop Down List. The selected Movie will display its category and the cost to stream the movie in separate labels. The application will then calculate the cost of the Movie (including taxes) in a separate form. The cost will vary according to the category of the movie as shown below. The user will be given the option to buy the movie and have a DVD copy sent to him for an additional charge of \$10.00 (including shipping).

Movie	Category
Season of the Witch	Sci-Fi
The Green Hornet	Action
The Dilemma	Comedy
Death Race 2	Action
Company Men	Drama
No Strings Attached	Comedy
The Way Back	Drama
The Mechanic	Action
The Rite	Horror
Sanctum	Action
The Other Woman	Action
The Roommate	Thriller
Waiting for Forever	Drama
Cedar Rapids	Comedy
Gnomeo and Juliet	Family
Just Go With It	Comedy
The Eagle	Action
I am Number Four	Sci-Fi
Footloose	New Release
Real Steel	New Release

Category	Download Cost
Comedy	\$1.99
Drama	\$1.99
Action	\$2.99
Sci-Fi	\$2.99
Horror	\$2.99
Thriller	\$1.99
Family	\$0.99
New Releases	\$4.99

File	Help
<u>P</u> rint	<u>A</u> bout
<u>S</u> tream	
<u>C</u> ancel	

Instructions:

1. Your App should contain 3 forms: **SelectionForm** (where the user selects the movie), **OrderForm** (where the user is shown the cost of the movie) and **StreamForm** (where the user is notified that his credit card has been charged and the video streaming is about to begin)

(25 Marks GUI, 20 Marks Functionality, 3 Marks: Program Structure, 5 Marks: Internal Documentation, 4 Marks Revision Control)

2. **SelectionForm (SubTotal: 7 Marks: GUI, 6 Marks: Functionality)**
 - a. Create a **Splash Screen** for the project that includes a company logo for "Movie Bonanza". Ensure that the form is displayed for a minimum of 3 seconds. (1 Mark: GUI, 1 Mark: Functionality)
 - b. Add a **drop down list** that displays a list of movies available to stream as shown in the above chart. Set the **Sorted** property to True. (1 Mark: GUI)
 - c. Add a **Group Box** to the form. The text of the **Group Box** will be set to "**Your Selection**". (1 Mark: GUI).
 - d. The Movie's **title**, **category** and **cost** will display in **3 text boxes** (along with appropriate labels) inside the **Group Box**. The textboxes will be set to "**Read Only**". (1 Mark: GUI, 1 Mark: Functionality)
 - e. Add a **Picture Box** to the **Group Box** which displays a **small graphic** of the movie that is selected (All graphics are provided with the assignment) (1 Mark: GUI, 1 Mark: Functionality).
 - f. Store the user's selection (including movie title, category and cost) in an **array** (2 Marks: Functionality).
 - g. Add a "**Next**" button that takes the user to the **OrderForm** and hides **SelectionForm**. This option will be grayed-out (disabled) until the user has selected a movie from the drop down list (1 Mark: GUI, 1 Mark: Functionality)
 - h. Set the **ControlBox**, **MaximizeBox**, **MinimizeBox** and **ShowIcon** properties of the form to False (1 Mark: GUI).
3. **OrderForm (SubTotal: 12 Marks: GUI, 11 Marks Functionality)**
 - a. Create a **Menu Strip** as shown above (1 Mark: GUI).
 - b. Add an **Cancel button** to the form (1 Mark: GUI).
 - c. The **Cancel menu selection** and the **Cancel Button** will close the Form using a **shared event** procedure (1 Mark: Functionality).
 - d. Add a "**Stream**" button to the form (1 Mark: GUI).
 - e. The "**Stream**" menu selection and the "**Stream**" Button will make the **StreamForm** visible and hide the **OrderForm** using a **shared event** procedure (1 Mark: Functionality).
 - f. The **Print Menu selection** will display a **print preview** of the **OrderForm**. (1 Mark: Functionality)
 - g. The **About Menu selection** will display an **AboutBox** containing the programmer's name, the version of the program and a website and contact number for the "Movie Bonanza" online movie streaming service (1 Mark: Functionality)
 - h. Add **2 group boxes** (one on the left side of the form and one on the right side of the form). The group box on the left will be labeled "**Movie Selected**" and the group box on the right will be labeled "**Your Order**" (1 Mark: GUI).
 - i. Add **2 text boxes** and a **Picture Box** (that will display a **larger graphic** of the user's movie selection) to the group box on the left that will display the user's selections from **SelectionForm** including the **Movie's Title** and the **Movie's Category**. The text boxes will be set to "**Read Only**" (1 Mark GUI: 1 Mark Functionality).

- j. Add **4 text boxes** (and appropriate labels) to the group box on the right that will display the **Movie's cost**, a **SubTotal**, the **Sales Tax** (at 13%), and a **Grand Total**. The text boxes will be set to **"Read Only"**. (3 Marks: GUI, 1 Mark: Functionality).
 - k. Add a **hidden label** and a **hidden textbox** to the group box on the right, positioned appropriately above the **SubTotal**. The text box will be used to display an additional charge of \$10.00 if the user decides to buy the movie and have the DVD sent to him (1 Mark GUI, 1 Mark: Functionality).
 - l. Add a **Checkbox** and to the group box on the right that will indicate the user has chosen to the buy the movie. When the checkbox is selected, the hidden label and hidden textbox will become visible on the form. Make sure to adjust the **SubTotal**, **Sales Tax** and the **Grand Total** text boxes (1 Mark: GUI, 2 Marks: Functionality).
 - m. Add a **"Back"** button to the form that takes the user back to the **SelectionForm** and hides the **OrderForm**. When the user is taken back to the **SelectionForm** his previous selection is displayed. (1 Mark: GUI, 2 Marks: Functionality)
 - n. Set the **ControlBox**, **MaximizeBox**, **MinimizeBox** and **ShowIcon** properties of the form to False (1 Mark: GUI).
- 4. StreamForm (SubTotal: 6 Marks: GUI, 3 Marks: Functionality)**
- a. Add an **"OK"** button to the form. When the **OK button** is clicked the form will **close** and the application will **terminate** (1 Mark: GUI, 1 Mark: Functionality).
 - b. Add a **label** that will thank the user for using "Movie Bonanza" (1 Mark: GUI).
 - c. Add a **label** that will notify the user that his credit card has been billed for the **Grand Total** as indicated in **OrderForm** (1 Mark: GUI, 1 Mark: Functionality).
 - d. Add a **label** that notifies the user that the Movie he has selected (from **SelectionForm**) will begin streaming in about a minute (1 Mark: GUI, 1 Mark: Functionality).
 - e. Set the **StartPosition** property of the form to **CenterScreen** (1 Mark: GUI).
 - f. Set the **ControlBox**, **MaximizeBox**, **MinimizeBox** and **ShowIcon** properties of the form to False (1 Mark: GUI).
- 5. Solution Structure (3 Marks: Program Structure):**
- a. Your solutions should include a Windows Forms named **SelectForm.cs**, **OrderForm.cs** and **StreamForm.cs** (1 Mark: Program Structure).
 - b. The Windows Form and all attached **UI Controls** must have appropriate Variable names with the following format: **ControlNameUIControlType** (e.g. **CalculateBMIButton**) (1 Mark: Program Structure).
 - c. Ensure all *private* class member variables (instance variables) use an underscore character (**_**) at the beginning of the identifier name to signify that they are private (or protected) (1 Mark: Program Structure).
- 6. Include Internal Documentation for your Application (5 Marks: Internal Documentation):**
- a. Ensure you include a **comment header** for your **C# files** that indicate: the **App name**, **Author's name**, **Student ID**, **App Creation Date**, **App description** (2 Marks: Internal Documentation).
 - b. Ensure you include a **section headers** for all of your **Event Handlers**, **Classes**, and any **functions** (1 Marks: Internal Documentation)
 - c. Ensure all your code uses **contextual variable names** that help make the files human-readable (1 Marks: Internal Documentation).

- d. Ensure you include **inline comments** that describe your GUI Design and Functionality. (1 Marks: Internal Documentation).
- 7. Share your files on **GitHub** to demonstrate Version Control Best Practices (**4 Marks: Version Control**).
 - a. Your repository must include **your code** and be well structured (2 Marks: Version Control).
 - b. Your repository must include **commits** that demonstrate the project being updated at different stages of development – each time a major change is implemented (2 Marks: Version Control).

FINAL FORM (SAMPLES)

1. SelectionForm

Movie Selection - Your Online Streaming Service

Choose the movie you wish to STREAM from the list below

Current Movies

- Cedar Rapids
- Company Men
- Death Race 2
- Footloose
- Gnomeo and Juliet
- I am Number Four
- Just Go With It
- No Strings Attached
- Real Steel
- Sanctum
- Season of the Witch
- The Dilemma
- The Eagle
- The Green Hornet

Your Selection

Title

Category Cost

Next

2. OrderForm

Your Order

File Help

Movie Selected

Title

Category

Your Order

Cost

SubTotal

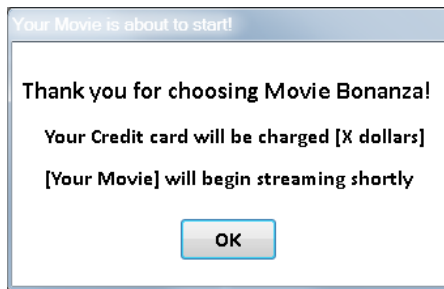
Sales Tax (13%)

Grand Total

☐ Order the DVD (adds \$10.00 to your order)

Back Cancel Stream

3. StreamForm



SUBMITTING YOUR WORK

Your submission should include:

1. A zip archive of your project files
2. A link to your project files on GitHub.

EVALUATION

Feature	Description	Marks
GUI / Interface Design	UI Controls meet the application requirements. Display elements are deployed in an attractive manner. Appropriate contrast is applied to application UI Controls and any background colours applied so that all text is legible.	25
Functionality	The program's deliverables are all met and the program functions as it should. No errors appear as a result of execution. User Input does not crash the program.	20
Program Structure	Your main "driver" class is named Program and it creates objects that are defined in other classes. All other classes are contained in their own files. Your classes use public properties and related private member variables wherever possible.	3
Internal Documentation	A program header is present and includes the name of the program, the name of the student, student number, date last modified, a short revision history and a short description of the program. All methods and classes include headers that describe their functionality and scope and follow commenting best practices. Inline comments are used to indicate code function where appropriate. Variable names are contextual wherever possible.	5
Version Control	GitHub commit history demonstrating regular updates.	4
Total		57

This assignment is weighted **10%** of your total mark for this course.

All Assignments are due at the beginning of class.

Late submissions:

- 20% deducted for each additional day.

External code (e.g. from the internet or other sources) can be used for student submissions within the following parameters:

1. The code source (i.e. where you got the code and who wrote it) must be cited in your internal documentation.
2. It encompasses a maximum of 25% of your code (any more will be considered cheating).
3. You must understand any code you use and include documentation (comments) around the code that explains its function.
4. You must get written approval from me via email.