# CIS605 Project 3

**Name:** **Sample Rubric**

* Note: Feedback may refer to a specific code line number ("LN"). To view line numbers in Visual Studio, navigate as follows: (Main Menu) > Tools > Options > (Options Box) > Text Editor > All Languages > check "Line Numbers"

| Topic | Expectations | Points Possible | Points Received | Specific Feedback |
|---|---|---|---|---|
| **General Functionality** *(applicable to all problem sets)* | | | | |
| Zip File, Solution, Projects | Program should be zipped properly and include the solution and all the applicable file(s) | Mandatory to receive a grade | | For all sections:<br>· Fix any issues from previous project submissions.<br>· Refresh comments for all code blocks that have changed.<br>· Ensure best practices have been applied to all new content too.<br>· Follow directions provided in the assignment and in class lectures. |
| Projects compile & bug-free | Each applicable project should compile cleanly (no compile errors, but compile warnings are sometimes okay) | -20% off of final point total | | |
| | Program should not crash while running | | | |
| **General Aesthetics** *(applicable to all problem sets)* | | | | |
| Look and Feel | UI elements align properly on form | 7 | 7 | |
| | UI elements appropriately sized/spaced | | | |
| | Good creativity while maintaining professionalism | | | |
| | User friendly captions and messages with proper spelling/grammar | | | |
| | Understandable ToolTips as appropriate | | | |
| | Message log entries scroll and display the most recent entry | | | |
| | Justification as appropriate (e.g. numbers are right justified) | | | |
| Usability | All tab stops correct | | | |
| | Keyboard shortcuts as appropriate | | | |
| | Appropriate Accept/Cancel functionality | | | |
| | Cursor reset to fix user input errors | | | |
| | Form reset after valid input | | | |
| **General Supportability** *(applicable to all problem sets)* | | | | |
| Template & Header | Official class template used in all files | 8 | 8 | |
| | Header completed for all files (LNs 3-15) | | | |
| | All code is in the proper template sections | | | |
| | Empty procedures are removed | | | |
| Internal Comments | All comments use clear business terms | | | |
| | Method comments exist | | | |
| | Section comments exist where longer blocks of code would benefit from them | | | |
| | Line-by-line code for very technical operations or where the code is not self-explanatory | | | |
| | End statements closed with a comment | | | |
| Naming convention | FrmMain and all other code files are named properly | | | |
| | UI elements are prefixed correctly and named in clear business terms (exception: elements not used in code, like many lables) | | | |
| | Variables and parameters are prefixed correctly and named in clear business terms | | | |
| | Methods and properties are named correctly | | | |
| Required Methods | _initializeUserInterface | | | |
| | _initializeBusinessLogic | | | |
| | ToString private and public override in classes | | | |
| Code Style | White space used effectively | | | |
| | Good logical blocks of code (e.g. local variables defined all together) | | | |

|  |  |  |  |  |
|---|---|---|---|---|
|  | Good separation of UI, Business Logic, and Data functions |  |  |  |
| **Required End User Functionality (Graded as the project is running)** |  |  |  |  |
| RUN TIME | **Correct use of test data:** | 10 | 10 |  |
|  | Button clicks on the content tabs |  |  |  |
|  | Process data button (hard coded tests) |  |  |  |
|  | **Functional Tabs:** |  |  |  |
|  | List boxes, combo boxes, populated with data consistent with test data and interactive data |  |  |  |
|  | **Summary Tab:** |  |  |  |
|  | List Boxes and Total counts should be populated. |  |  |  |
| **Required Code Elements (Graded as a code review)** |  |  |  |  |
| FrmMain | **Button clicks:** | 12 | 11 | 1 Only one module variable ThemePark varaible be used in FrmMain in all subs/Functions to maintain the consistency of the project, the correct count of objects, and the created EventArgs objects on the event of creating a new objects, such as using this local variable in the process test data, Dim themePark As ThemePark |
|  | Validate input fields with if/then and try/catch |  |  |  |
|  | Call behavioral methods in ThemePark |  |  |  |
|  | Updates to the form are made only by responding to custom events that are raised in the ThemePark behavioral methods. |  |  |  |
|  | **Hard coded test data:** |  |  |  |
|  | Appropriate objects used |  |  |  |
|  | Call behavioral methods in ThemePark |  |  |  |
|  | Updates to the form are made only by responding to custom events that are raised in the ThemePark behavioral methods. |  |  |  |
|  | **Required Custom Event Handlers:** |  |  |  |
|  | Listens for events from ThemePark |  |  |  |
|  | Adds information to all applicable lists, combos, textboxes, and labels |  |  |  |
|  | Updates the transaction log. |  |  |  |
| ThemePark | **Correct attributes and properties** | 10 | 10 |  |
|  | Total counts for each object |  |  |  |
|  | **Methods:** |  |  |  |
|  | Specified business process methods |  |  |  |
|  | Creates other primary objects in the process methods (e.g. AddCustomer creates the Customer object) |  |  |  |
|  | Calculations as possible |  |  |  |
|  | Correct parameters and logic |  |  |  |
|  | **Custom Events:** |  |  |  |
|  | Use proper EventArgs |  |  |  |
|  | Fire at appropriate places |  |  |  |
|  | Structured correctly |  |  |  |
| Classes | Correct attributes and properties | 10 | 10 |  |
|  | Constructor |  |  |  |
|  | Behavioral Methods |  |  |  |
|  | ToString |  |  |  |
|  |  |  |  |  |
| EventArgs Classes | Created for every data transaction | 18 | 15 | 3 The requirment in the project 3 document to create 6 EventArgs classes, Class ThemePark_EventArgs_PassbookFeaturePurchased is missed, which is used to create a new PassbookFeature object; this is different from Class ThemePark_EventArgs_PassbookFeatureUpdated that is used to updated PassbookFeature object, in this project this class will create a new PassbookFeature object, which will update PassbookFeature object in project 4 |
|  | Inherits System.EventArgs |  |  |  |
|  | Correct attributes and properties |  |  |  |
|  | Constructor |  |  |  |
|  | ToString |  |  |  |
|  |  |  |  |  |
| **Overall Feedback and TOTAL** |  |  |  |  |
| **Sample Rubric** |  | **75** | **71** |  |