# Emma's Small Engine Case Study

## Programming Objectives:

- Develop an ASP.NET web solution for the Emma's Small Engine business case from PROG1180.
- Use ADO objects and programming techniques from PROG1210 to implement the database functionality of the new Emma's Small Engine system.

## Project Objectives:

- Delegate project tasks fairly and in a manner that supports the strengths of each team member.
- Communicate progress and delays in assigned tasks for each deliverable in a timely fashion.
- Perform reviews of peers' project work.
- Provide constructive feedback when reviewing peer work.
- Coordinate peer review comments into project work.
- Consult with various team members, before and after a project deliverable, to identify personal areas of growth.

## Project Requirements:

This project assignment is based on the PROG1180 Case Study and is completed in teams. In this case you must develop a new system for **Emma's Small Engine.** Your team will create an **ASP.NET Web Prototype** that contains web pages to represent the entire system including a main landing page and a log in page. All the web pages in your solution must be fully designed and the navigation between the pages must be functional. Your solution should model the analysis (Story Board) and design work from PROG1180. A database is provided. Customize this Database as necessary.

## Required Coding TODO:

You must code the pages that pertain to **Customers** and their **Purchases** and **Repairs**. Add functionality to maintain customer records including adding new customers to the system and for processing their purchases and repairs. Provide a user-friendly search feature for locating **Orders** and displaying relevant information to complete **Deliveries**.

## Required Reporting TODO:

Design an administrative reporting page that provides Emma options to query her **Customers** and **Sales**. Provide information for **Decision Making Support** and information that will help Emma appreciate her **Employees**.

**Note:** You are **not** required to code the **Inventory** or **Ordering** portions of this system. You do not have to code pages that maintain Lookup tables. Instead these pages are Designed and 'Under Construction'.

## Marks:

This project is worth 25% of your overall course mark. This project consists of a **Prototype Meeting** worth 10% of your final grade. A brief **Presentation** and **Final Product** is submitted at the end of this term and is worth 15% of your final grade. See the rubrics below for details on these milestones.

## Preliminary Work:

As a team decide on the professional naming conventions and folder organization of your assemblies and files. Use a class Library for data access classes, methods and Datasets. Remember namespaces, classes, and methods are uppercase. Fields and variables are lowercase. All identifiers should follow camel notation. All code files should be documented with team name/members who contributed to each source file. The first name listed is the primary student responsible for that source file. Add names to the top of designer generated files created thru RAD techniques such as Dataset files.

## Design Instructions:

- Make sure you transfer your design concepts from your work in PROG1180. Come up with design standards for your team to follow: colour schemes, naming conventions, form real estate issues etc.
- Each form/webpage that a user can access through your system navigation should be created to illustrate where all of the data input and/or output controls will appear on the form. This layout will include the sequence and groupings of controls on a form/page. In addition, your team's designs will show the type of controls that will be used to gather or output data. For example, will a user enter a particular item into a text box; select it from a list box, combo box, or option group etc.? Lastly, the forms should include your group's "creative take" on the colours, fonts, and graphics that will be used to enhance the look, feel, and more importantly, the functionality of the System.
- In your design, include a consistent method for indicating the required fields on each form/page (i.e. back colour, red asterisk, bolding etc.).
- The overall system is consistent, easy to navigate (i.e. choice of hotkeys, logical flow, all options for a function is one click away) Initial bullet-proofing guides the user through transactions in a logical sequence, controls add/edit/delete functions and navigation. There is an effective use of controls for gathering and/or outputting data.
- The use of colour, fonts, graphics, and the spacing and grouping of controls enhance the overall functionality of the system.
- **Optional Bonus**: The web is responsive and incorporates accessibility standards.

## Coding Instructions:

- Develop required crud operations using ADO objects and programming techniques from PROG1210.
- Search customers, repairs, and orders providing the user with different ways to filter and locate records.
- When a customer record is selected the user should be able to see a list of existing orders and repairs for that customer.
- Provide a way to maintain customer records and to process new orders, repairs and manage deliveries.
- Build a functional administrative page for Emma that provides useful information about employees, customers, orders, and repairs.
- Correct all code to ensure the site is functioning as required.
- Add code to ensure the web site is user-friendly and robust. Include data validation, exception handling, and status and error messages.

## Prototype Meeting:

- Create an ASP.NET web site based on your analysis and design work from PROG1180. Ensure all web forms are user friendly and incorporate a consistent design throughout by applying the design concepts you have learned in previous courses (CSS, JavaScript, and HTML).
- Add a login feature for staff; only a valid login (username and pass word) should provide access to the site beyond the landing page.
- Ensure the user can navigate all site pages.

**Prototype Meetings** are scheduled during regular class time in **Week 12**. A schedule of specific times and location for each team will be posted in the Blackboard Weekly folder. The lab will be available for students to work on their projects and assignments.

All team members must attend the prototype meeting in order to receive their individual mark. Be on time, prepared, informed, and well organized. All team members must share equally in the meeting, and incorporate feedback into the final submission. All members must submit a hard copy **Peer Evaluation** Form. The team must submit a hard copy of the **Team Work Assignment Breakdown** form.

**Marking Rubric**

| *Category* | 1 | 2 | 3 | Points |
|---|---|---|---|---|
| Web Interface | The resulting web site has some key pages, but is missing many other necessary pages which would be required to support this system. | Most pages are present in the design to support minimum functionality. Only a few edits are necessary to arrive at a complete site. | All require web pages are present in the site to support minimum functionality. | *5 = /15 |
| Web Design | Produced a standard design/template; implementation of design principles needs improvement. | Product incorporates an interesting design; acceptable implementation of design principles. | Produced a unique and creative product; successfully applied design principles. | *5 = /15 |
| Web Navigation | The navigation is missing for many pages; not clear how the site would work. | Most of the navigation is working. Some edits required to demonstrate how some parts would work. | All page navigation is present. It is clear how the site would work. | *5 = /15 |
| Login | The resulting web site has some login operations working, but is missing what is required to support the system | Most of the login operation is present in the site to support minimum login. Only a few operations are necessary to arrive at a complete login. | Login capability is present in the site and working correctly. | *5 = /15 |

## Final Submission:

Submit one final copy of your team's solution to this case study in your team file share for marking:

**Presentation** presented during regular scheduled class of PROG1210 in **Week 15**. The team must select at least one member to showcase their solution in a brief presentation with the class present. All team members must attend the presentation in order to receive their individual mark.

**Due Date: Friday, December 14, 2018, 4:30pm**

## Marking Rubric

| Category | 1 | 2 | 3 | Points |
|---|---|---|---|---|
| **Implementation of expected system** | The resulting assemblies have some design concepts but are missing most which would be required to support the system. | The analysis and design concepts are present to support the system. Only a few edits are necessary to arrive at a complete product. | All analysis and design concepts are present in this system. | *5 = /15 |
| **Site Function** | The resulting web site has operations working, but is missing many which would be required to support the system | Most operations are present in the site to support minimum functionality. Only a few operations are missing to arrive at a complete solution. | All functions required are present in the site to support the required functionality. | *10 = /30 |
| **Search Operations (locate records)** | The resulting web site has some search operations working, but is missing many which would be required to support the system | Most search operations are present in the site to support minimum functionality. Only a few operations are necessary to arrive at a complete solution. | All search operations are present in the site to support the required functionality. | *5 = /15 |
| **CRUD Operations** | The resulting web site has some crud operations working, but is missing many which would be required to support the system. Data Integrity is at risk. | Most crud operations are present in the site to support minimum functionality. Only a few operations are necessary to arrive at a complete solution. | All crud operations are present in the site to support the required functionality and to maintain data integrity. | *5 = /15 |
| **Reporting** | The resulting assemblies have some reporting but are missing information which would be required to support decision making. | Most required reporting is present in the design to support decision making. Only a few edits are necessary. | All required reporting is present in the design to support decision making. | *5 = /15 |
| **Professional Standards** | Naming conventions and organization are inconsistent and/or rarely follow program naming conventions or system naming context. | Naming conventions and organization follow program standards. More detail is required to meet standards and system naming context. | Naming conventions and organization follow program standards and are descriptive/reader friendly. | *2 = /6 |
| **Documentation** | Documentation is missing and lacks organization | All documentation is submitted, organized and reader friendly. | N/A | *2 = /4 |