QKM2 — QKM2 TASK 1: OBJECT-ORIENTED APPLICATION DEVELOPMENT

SOFTWARE I — C482
PRFA — QKM2

TASK OVERVIEW SUBMISSIONS **EVALUATION REPORT**

EVALUATION REPORT — ATTEMPT 2 — PASSED

Overall Evaluator Comments

EVALUATOR COMMENTS

The deletion confirmation, inventory boundaries and product-with-part deletion exceptions perform well. Congratulations on completing the last aspects and passing the assessment.

A1. Main Form

Competent The submitted Main form closely matches the organization of the GUI layout and includes *all* required components. The code uses a permitted tool to create the GUI, does not contain errors, and is complete.

There are no comments for this aspect.

A2. Add Part Form

Competent The submitted Add Part form closely matches the organization of the GUI layout and includes *all* required components. The code uses a permitted tool to create the GUI, does not contain errors, and is complete.

There are no comments for this aspect.

A3. Modify Part Form

Competent The submitted Modify Part form closely matches the organization of the GUI layout and includes *all* required components. The code uses a permitted tool to create the GUI, does not contain errors, and is complete.

There are no comments for this aspect.

A4. Add Product Form

Competent The submitted Add Product form closely matches the organization of the GUI layout and includes *all* required components. The code uses a permitted tool to create the GUI, does not contain errors, and is complete.

There are no comments for this aspect.

A5. Modify Product Form

Competent The Modify Product form closely matches the organization of the GUI layout and includes *all* required components. The code uses a permitted tool to create the GUI, does not contain errors, and is complete.

There are no comments for this aspect.

B. Javadoc Comments

Competent Javadoc comments are provided for *each* class member throughout the code and include detailed descriptions of a logical or runtime error corrected in the code, how the error was corrected, and a description of a future enhancement that would extend the functionality of the application if it were to be updated.

There are no comments for this aspect.

C. Class Structure

Competent The application maps to the UML class diagram and includes *all* classes and members shown in the diagram. The code correctly demonstrates *each* given point, and includes the unmodified Part class that is provided.

There are no comments for this aspect.

D1. Parts Pane

Competent *Each* button on the Parts pane functions properly. The application displays matching search results in the Parts TableView if the part or parts are found, or the application displays an error message in the UI or in a dialog box if the part or parts are not found. The empty search field functions correctly.

There are no comments for this aspect.

D2. Products Pane

Competent *Each* button on the Products pane functions properly. The application displays matching search results in the Products TableView if the product or products are found, or the application displays an error message in the UI or in a dialog box if the product or products are not found. The empty search field functions correctly.

There are no comments for this aspect.

D3. Exit Button

Competent The Exit button functions properly by closing the application.

There are no comments for this aspect.

E1. Add Part Form Functionality

Competent The Add Part form includes correctly functioning In-House and Outsourced radio buttons, auto-generates a unique part ID but leaves the part ID text field disabled, and allows users to enter the given part details into their corresponding active text fields. Users are automatically redirected to the Main form after saving data or after canceling or exiting the form.

There are no comments for this aspect.

E2. Modify Part Form Functionality

Competent The Modify Part form includes populated text fields with the data from the chosen part. The form includes correctly functioning In-House or Outsourced radio buttons and retains the part ID when the Save button is clicked. Users can modify existing values in the text fields except the part ID and are automatically redirected to the Main form after saving modifications to the part or after canceling or exiting the form.

There are no comments for this aspect.

F1. Add Product Form Functionality

Competent The Add Product form auto-generates a unique product ID. The auto-generated product ID populates but is disabled. The user can enter the given product details and search for parts by ID or name. The application displays matching search results for a single part or filters multiple parts if they are found, or the application displays an error message in the UI or a dialog box if the part or parts are not found. The empty search text field functions correctly, the top table is identical to the Parts TableView in the Main form, the Add button copies the selected part to the lower table, the Remove Associated Part button functions as required, and users are automatically redirected to the Main form after saving modifications to the part or after canceling or exiting the form.

There are no comments for this aspect.

F2. Modify Product Form Functionality

Competent The Modify Product form includes text fields populated with data from the chosen product, and the bottom TableView populates with the associated parts. The form allows the user to search for parts by ID or name and displays matching search results for a single part or filters multiple parts if they are found, or it displays an error message in the UI or in a dialog box if the part or parts are not found. The empty search text field functions correctly, the top table is identical to the Parts TableView in the Main form. The user can modify data values, and the product ID is disabled. The user can associate, remove, or disassociate a part from a product. Associated parts are selected from the top table and are stored on the bottom table. Users are automatically redirected to the Main form after saving modifications to the part or after canceling or exiting the form.

There are no comments for this aspect.

G. Input Validation and Logical Error Checks

Competent The code implements input validation and logical error checks for *each* of the given circumstances and displays a descriptive error message when an error is detected.

EVALUATOR COMMENTS: ATTEMPT 2

The deletion confirmation, inventory boundaries and product-with-part deletion exceptions perform well.

H. Javadoc Folder

Competent The submission includes a folder containing Javadoc files that were generated from the IDE or via the command prompt from part B and specifies, in a comment above the main method header declaration, where the folder is located.

There are no comments for this aspect.

I. Professional Communication

Competent Content reflects attention to detail, is organized, and focuses on the main ideas as prescribed in the task or chosen by the candidate. Terminology is pertinent, is used correctly, and effectively conveys the intended meaning. Mechanics, usage, and grammar promote accurate interpretation and understanding.

There are no comments for this aspect.