Student Management Hub

- **Problem Statement:** How can a student manage his/her grades and schedule with a user-friendly digital tool?
- **Project Objectives:**
    - Students access a digital application that recognizes and helps them monitor and organize their grades or schedules through a menu interface.
    - Students can enter their subject grades allowing the application to predict their GWA (General Weighted Average).
    - Students have a choice to enter their schedules in an orderly date loop as a result of giving an output after or enter their daily tasks then type “Done” after doing a task.
- **Planned Features:**
    - Log-in interface:
        - for creating/opening a student account
    - Menu interface:
        - choice to open student grades management or weekly schedule
    - Student Grade Management:
        - computes your general weighted average for the quarter
    - Weekly Schedule has options to:
        - track your tasks and cross them out once done
        - add due dates by week or by month
    - Data storage:
        - student information will be packed into a single file per student account, which may then be opened by a copy of the program on another device
- **Planned Inputs and Outputs:**
    - **Inputs:**
        - Student account information for the login interface
        - Choices for menu interfaces
        - Student Grade Management:
            - Student scores for quarterly grade calculation
        - Weekly Schedule:
            - Task information and due dates
    - **Outputs:**
        - Displays student account once made in the log-in interface
        - Directs the user to open either the Student Grades Management or the Weekly Schedule option
        - Student Grades Management:
            - Outputs calculated General Weighted Average
        - Weekly Schedule:
            - Outputs tasks left to do (to-do list)
            - Displays the final organized calendar
- **Logic Plan:**

```
DEFINE gradeManagement():
    DECLARE INTEGER choice, month, day
    DECLARE FLOAT score, maximum
    DECLARE STRING name, date

    WHILE True:
        OUTPUT “1. Enter grades for the quarter”
        OUTPUT “2. Transmute quarterly averages”
        OUTPUT “3. Exit”
            INPUT choice
            CASE choice OF:
                1:
                    WHILE True:
                        OUTPUT “1. Log an FA”
                        OUTPUT “2. Log an AA”
                        OUTPUT “3. Select an activity from the calendar”
                        OUTPUT “4. Finish logging”
                        OUTPUT “Enter the number corresponding to your
                        choice… “
                        INPUT choice

                        CASE choice OF:
                            1:
                                OUTPUT “Enter the subject:”
                                OUTPUT “Enter the name of the
                    activity… “
                                INPUT name
                                OUTPUT “What’s the maximum
                    score for this? ”
                                INPUT maximum
                                OUTPUT “What’s your score on this
                    activity? “
                                INPUT score
                                [Data is written to a file. Activity is
                    labelled as an FA.]

                            2:
                                OUTPUT “Enter the subject:”
                                OUTPUT “Enter the name of the
                    activity… “
                                INPUT name
                                OUTPUT “What’s the maximum
                    score for this? ”
                                INPUT maximum
```

```
                    OUTPUT “What’s your score on this activity? “
                    INPUT score
                    [Data is written to a file. Activity is labelled as an AA.]

                3:
                    OUTPUT “Enter the month (01 - 12): ”
                    INPUT month
                    OUTPUT “Enter the day (01 - 0#)[1]: “
                    INPUT day
                    [Using the given data, the calendar file is searched through for the activities due on the given date.]
                    OUTPUT “Select the activity you wish to log…”
                    [A list of the activities due on the entered date is printed with numerical indexes.]
                    INPUT choice

                4: break
                default : OUTPUT “Please enter a valid input.”

        2:
            [Quarterly activities are read from the calendar file.]
            [The PSHS grade formula is used to compute the quarterly average.]
            [The raw average is then transmuted using the previous quarter’s average. (In the first quarter, it is left as it is.)]

        3:
            break

        default:
            OUTPUT “Please enter a valid input.”


DEFINE scheduler():
    DECLARE INTEGER choice
    DECLARE STRING name, date
```

[1] The stop value depends on the selected month, hence the place-holding "#".

```
        WHILE True:
                OUTPUT “1. Log a deadline”
                OUTPUT “2. Finish logging”
                OUTPUT “Enter the number corresponding to your choice… “
                INPUT choice

                CASE choice OF:
                        1:
                                OUTPUT “Enter the name of the activity… ”
                                INPUT name
                                OUTPUT “When is this due? (mm/dd/yy) “
                                INPUT date
                                [Data is written to a file.]

                        2:
                                break

                        default:
                                OUTPUT “Please enter a valid input.”

DEFINE main():
        DECLARE INTEGER choice
        OUTPUT “Enter 1 for Grade Manager and 2 for Weekly Scheduler: ”

        WHILE True:
                INPUT choice
                IF choice is equal to 1:
                        CALL gradeManagement()
                ELSE IF choice IS EQUAL TO 2:
                        CALL scheduler()
                ELSE:
                        OUTPUT “Please enter a valid input.”
```