***Welcome to week 9 Group Activity:***

***Lead🡺 Tinotenda…….***

***Opening Prayer🡺 Any volunteer***

**Things to discuss**

**Unit 05 Develop: Eternal Quest Program**

***Step 4 : Review the Program Specification***

Refer to the Eternal Quest program specification. As a team, review the program requirements and how it is supposed to work.

1. What does the program do?

2. What user inputs does it have?

3. What output does it produce?

4. How does the program end.

***Co-requirements for the program***

1. **What does the program do?**

\*\*1 displays on opening menu

\*\*2 provides goals for running a marathon, reading scriptures and visiting the temple.

\*\*3 displays opening message

\*\*4 asks the user to create new goals of any type.

\*\*5 starts the program and leads the user through it with prompts

\*\*6 asks the user to record an event.

\*\*7 diplays the user’s score.

\*\*8 displays closing message

1. **What user inputs does it have?**

\*\*1 user’s score.

\*\*2 choosing a goal

\*\*3 answers to prompts

\*\*4 recording events

1. **What output does it produce?**

\*\*1 menu

\*\*2 opening message

\*\*3 prompts

\*\*4 scores

\*\*5 closing message

1. **How does the program end?**

\*\*1. at the end of program, it tells the user the total score of the goals completed as for a closing message. Tells what goals they completed and how long they did it. returns to the main menu

\*\*2 end of program enter quit to end the program

***Step 5: Determine the classes***

\*\*1) **Menu class**: to display the menu

\*\*2) **program class**: to display the beginning and ending messages

\*\*3) **Running a marathon class**: to run the marathon goal

\*\*4) **Reading scriptures class**: to run the reading scripture goal

\*\*5) **Visiting the temple class**: to run the visiting the temple goal

***Steps 6: Define class behaviors , 7: Define class attributes &***

***8: Define Constructors***

#1\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Class: **Menu (- means private while + means public)**

Attributes:

* \_programList : List<Activity>

Constructors:

+ Menu()

Behaviors:

+ DisplayMenu() : string

How it will work:

The Menu class will display the choices for the user for the program and record that choice as according to what the user chooses.

#2\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Class: **Program**

Attributes:

* \_name : string
* \_openingMessage : string
* \_closingMessage: string
* \_scores : int

Constructors:

+ Program(string name, string openingMessage, string closingMessage)

Behaviors:

+ DisplayMessage(string message) : void

+DisplayScores(): void

How it will work:

It provides the full detailed about the program and does the opening and closing messages as well as showing the scores and other behaviors that are common in the derived program classes.

#3\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Class: **MarathonProgram**

Attributes:

* \_promptMarathon : List<string>

Constructors:

+ MarathonProgram()

Behaviors:

+ RunningMarathon() : void

+ NotRunningMarathon() : void

How it will work:

It shows and records if the program is done and completed well.

It runs prompts with a specific score to be obtained after finishing the goal.

#4\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Class: **ReadingScripturesProgram**

Attributes:

* \_promptList : List<string>
* \_scriptureList: List<string>
* \_relatedscriptures: List<string>

Constructors:

+ ReadingScripturesProgram()

Behaviors:

+ ScriptureRead() : string

+ DisplayScriptureRead() : void

+ RelatedScriptures(): string

+ DisplayRelatedScriptures: void

How it will work:

It records the user’s scripture read from the \_scriptureList and display it. Lists scriptures that are related to the original prompt.

#5\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Class: **VisitingTempleProgram**

Attributes:

* \_promptList : List<string>
* \_visitingtempleList: List<string>

Constructors:

+ VisitingTempleProgram()

Behaviors:

+ DisplayPrompt() : void

+ UserList() : int

+DsiplayVisitingtemple() : void

+TimesVisitedList() : int

How it will work:

It shows a number of visits to a temple of thye user from the \_visitingtempleList so the user gets to see how many times visited. Tracks the number of visits made as the user presses enter after each response. Reports the number of responses at the end of the activity.

#6\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Class: **Scores**

Attributes:

* \_name : string
* \_points : int

Constructors:

+Score()

Behaviors(methods):

+ Points() : int

+ DisplayPoints() : void

How it will work:

Displays the user’s total scores in points.