



Software Development Teach Yourself Series

Topic 1: Problem Solving Methodology Units 1-4

A: Level 14, 474 Flinders Street Melbourne VIC 3000
T: 1300 134 518 **W:** tssm.com.au **E:** info@tssm.com.au

Contents

As it appears in Units 1-4.....	4
Analysis.....	5
1. Data Collection.....	5
2. Analysis of Data.....	5
3. Determine the solution requirements	7
4. Identify the constraints of the solution.....	7
5. Determining the scope of the solution	8
Analysis Review Questions	9
Design	10
Design Tools	10
Interface Design	10
Data Dictionary	11
Object Description Table	11
Pseudocode.....	11
Evaluation Criteria	12
Design Review Questions	12
Development	13
Manipulation of data	13
Validation.....	13
Testing.....	13
Documentation	13
Review Questions	14
Evaluation	15
Determining a strategy	15
Reporting.....	15
Review Questions	15
Solutions to Review Questions	16
Analysis.....	16
Design	17
Development	18
Evaluation	18

CASE STUDY



All Teach Yourself Series in this package will refer to the following case study.

Tariq Mulner is the manager of a school canteen. He manages how many lunches are going to be prepared each day. It is difficult to tell how many lunches will be sold, so he would like a software solution that students can use to order lunches. This application would provide him with a complete list of orders.

Most of the students have smart phones, so Tariq is suggesting the solution is a phone app that can read in the lunch order, and send it to his device so he can print out the order list.

Photo by Tirachard Kumtanom from Pexels used with permission

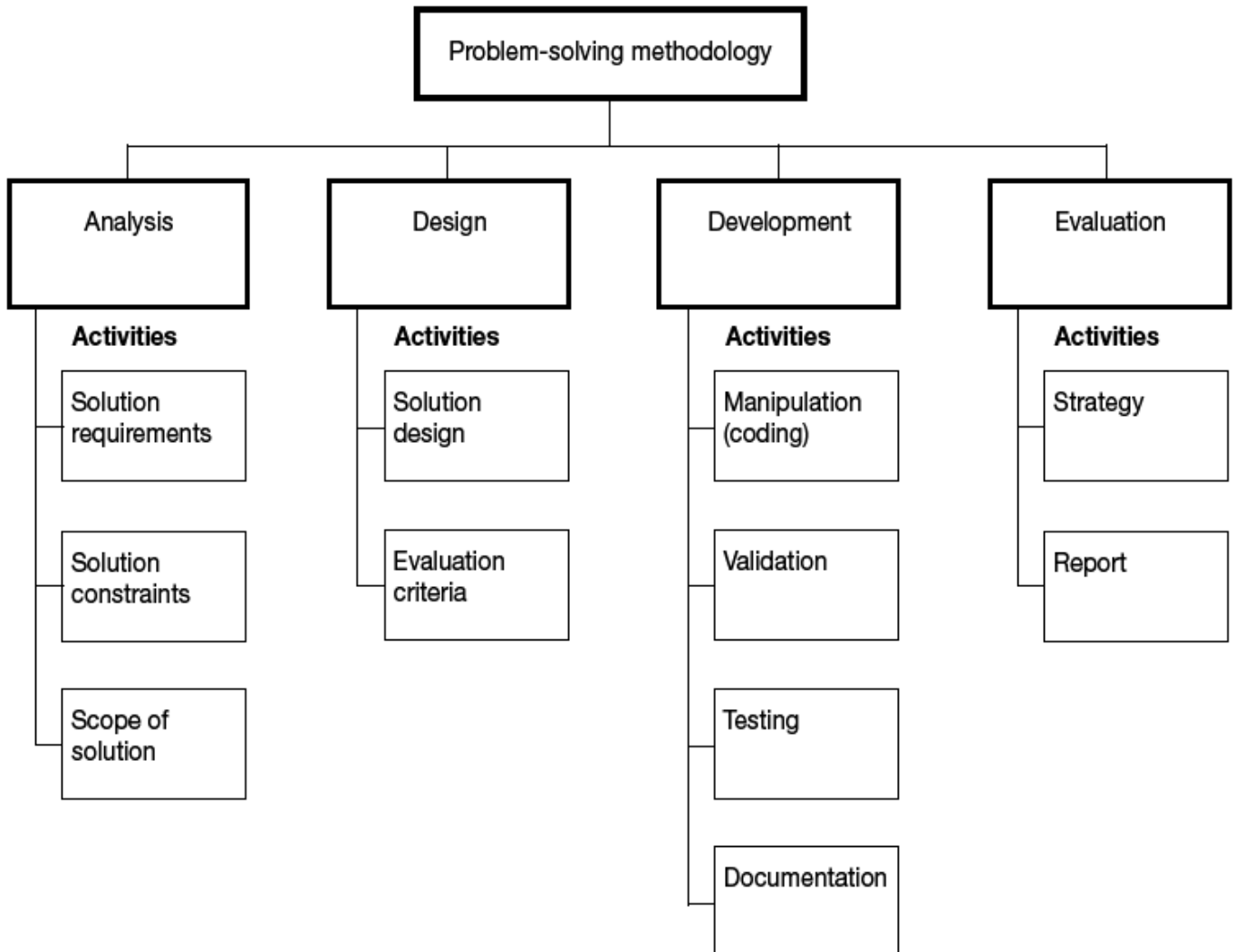


Problem Solving Methodology

As it appears in Units 1-4

The problem-solving methodology is used throughout the Computing study design. It refers to the steps taken to solve an information problem.

There are four stages of the problem-solving methodology. Each stage is made up of activities.



Source: VCAA Study Design

Analysis

Analysis is the first stage of the problem-solving methodology. Analysis involves the investigation that must be completed before an information problem can be resolved. What is the problem the developer needs to solve? What benefits will the solution provide the user?

1. Data Collection

The developer needs to collect data from stakeholders. There are a number of ways of collecting data:

- Observations
- Focus groups
- Surveys & Questionnaires
- Interviews
- Data mining online data sets and repositories
- Researching secondary sources such as publications, textbooks and reports.

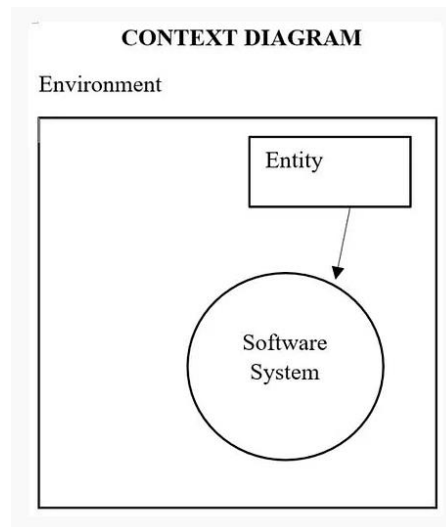
2. Analysis of Data

Once the data has been collected, analysis can take place using the following analysis tools:

- IPO Chart
- Context Diagram
- Use Case Diagram
- Data Flow Diagram

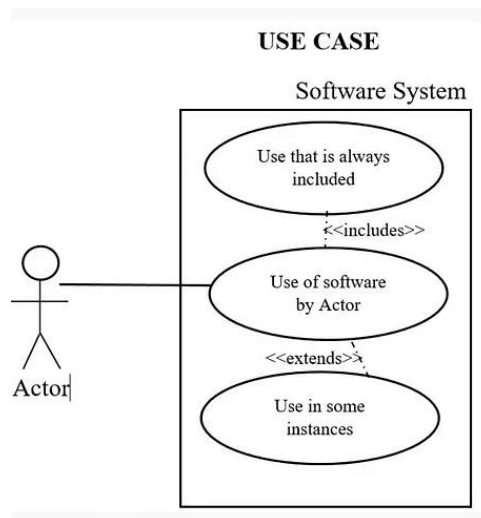
Context Diagram

Context Diagrams are used to show the software in context of where they will be used.



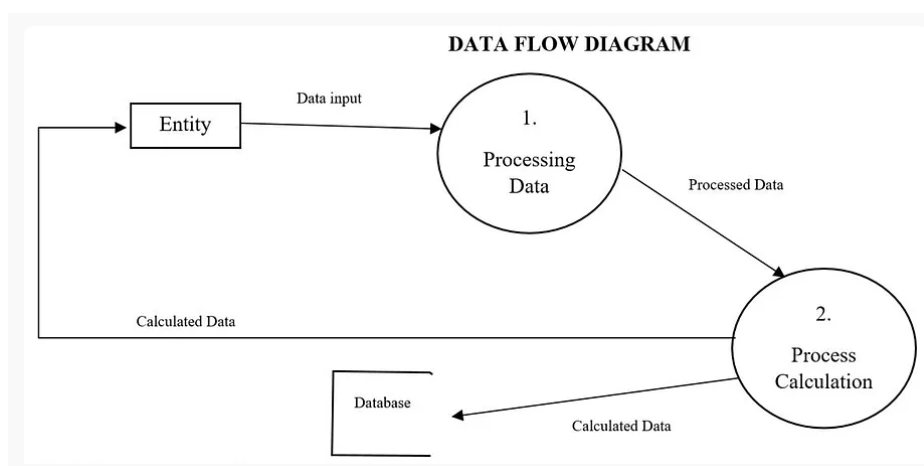
Use Case Diagrams

These diagrams are used to illustrate all the elements of the software's use.



Data Flow Diagram

The most complex of the Analysis tools, the DFD illustrates how the data flows through the proposed software and all the processes required.





In the case of Tariq's Canteen App, it is important to identify the requirements of the software within the scope and constraints.

3. Determine the solution requirements

Once the problem has been analysed, we can now list the requirements. There are two types, functional and non-functional.

- Functional: what the solution is required to do
- Non functional: the attributes of the solution such as being user friendly, reliable, portable, robust, or maintainable.



Functional Requirements for Tariq's Canteen App:

- Must enter student Name and ID number to login
- User to be able to search for lunch items by type
- Sends lunch order to database
- Must be able to print out all the lunch orders
- Create a list of total numbers required for each lunch item

Non-Functional Requirements for Tariq's Canteen App:

- Easy to use on a mobile phone
- Robust enough to be able to deal with error correction when ordering
- Reliable list of orders and totals

4. Identify the constraints of the solution

Constraints refer to the restrictions that need to be considered when designing a solution. Constraints may include:

- Cost
- Speed of processing
- Requirements of users
- Legal requirements
- Security
- Compatibility
- Level of expertise
- Capacity
- Availability of equipment



In Tariq's case the constraints will include the processing and storage limitations of a mobile phone. There is a deadline for the beginning of school term to implement the app to evaluate it. Security will be important as it processes student ID Numbers.

5. Determining the scope of the solution

Scope refers to the boundaries or parameters of the solution. Scope should consider:

- What the solution can do
- What the solution cannot do



Tariq requires the application produce a report that identifies the number of each lunch item ordered. It will also need to allow Tariq to search for a specific order in case there is an error. There is a possibility that the solution could monitor the stocktake of ingredients and items within the kitchen and process order to wholesalers. In this prototype – the stocktake is not included.

Design

The design stage involves determining the way the problem will be solved.

During this stage it may be necessary to create:

- Mock-up diagrams of interface or visualisation
- Data Dictionaries
- Object Description Tables
- Pseudocode

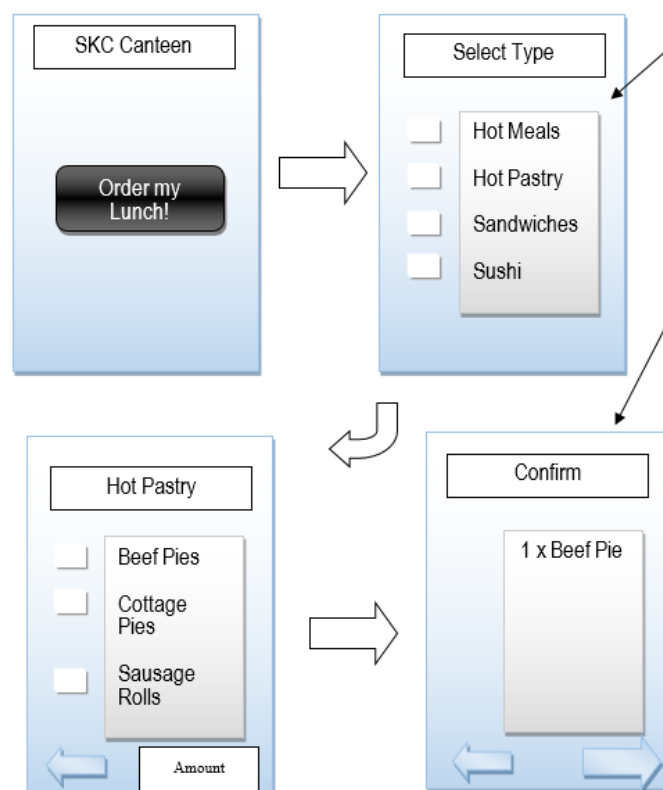
The solution design involves:

- Identifying the data that is required
- Determining how the data will be named, structured, validated and manipulated
- Showing how each part of the solution relates to other parts
- Determining the appearance of information generated from a solution
- Determining the evaluation criteria

Design Tools

Interface Design

When designing an interface, it is important that it annotated with descriptions of its functionality, and appearance.



Design A: A simple, but an unappealing screen design, which may not be easy to maintain. With Check boxes, Mr. Philips will not be able to easily update the products. This is a functionality issue limited by the skill level on the Manger Mr. Phillips.

Back Buttons not clear this is a accessibility and readability issue that needs to be fixed.

There needs to be a Cancel button to allow for incomplete orders to be re-set. Possibly a Cart form.

Again the price setting in the code is not ideal for maintainability. For the Main user, the product prices need to be updated from time to time which need to be in an easy to use format such as a text file.

Data Dictionary

A data Dictionary identifies all the variables and data structures used in the software solution. The variable Name and data type need to be listed. The size is in bytes. Scope is Local or Global which determines how the variable is used within the solution.

Variable Name	Data Type	Size	Scope	Description
intAmount	Integer	1	Global	Handles the amount of a product ordered.
dblPrice	Double	4	Global	Handles the individual price of a product chosen
dblTotalCost	Double	4	Global	Total Price of the order

Object Description Table

This table lists the names and object types used in the interface of the software. There are two types of objects: **Event – triggers code**, and **Method – edits properties**.

Object Type	Object Name	Type	Variable Handled	Description
Form	frmHome	Event	N/A	This is the home screen that starts the app.
Form	frmTypes	Event	N/A	This is the screen provides the meal type options
Button	btnStart	Event	N/A	Opens frmTypes
CheckBox	chbHMeal	Method	Boolean	Selects Hot Meals
CheckBox	chbHPastry	Method	Boolean	Selects Hot Pastry
CheckBox	chbSandwich	Method	Boolean	Selects Sandwiches
CheckBox	chbSushi	Method	Boolean	Selects Sushi
Textbox	txtAmount	Method	intAmount	Reads in the Amount of the products

Pseudocode

Pseudocode is the type of code used to design algorithms in Software Development. This is where the developer solves the logic problems of the software solution.

The features include:

- Every algorithm must begin and end with START and STOP or BEGIN and END.
- When assigning data to a variable use ←. Example: IntNumber f5
- Show each calculation required. Example: IntAnswer←IntNumber1 + IntNumber2
- Decision Control Structures. Example: IF condition THEN Action1 ELSE Action2 ENDIF
- Case Control Structures. Example: Case where condition, CASE1 Action1 CASE2 Action2 END CASE
- Repetition Counted Loop Control Structures. Example: FOR counter ←first value to last value END FOR
- Repetition Post-Test Loop Control Structures. Example: REPEAT actions UNTIL condition
- Repetition Pre-Test Loop Control Structures. Example: WHILE condition DO actions END WHILE
- Use indentation and spaces to illustrate how control structures are incorporated
- Conditions: ==(equal), <> (not equal), > (greater than) and < (less than).

Evaluation Criteria

Once a analysis has been completed it is important to produce a list of evaluation criteria based on the requirements. We do this before the development stage.

No	Requirement	Strategies	Effectiveness & Efficiency
EC01	(FR01) Interface of the Customer's App allows a choice selection from all products available. Does the interface allow the full selection of available products?	Client to provide all choice selections are available in documentation data. Client to check if all selections have been made available in the interface. During Desk Checking and Final System Checking all sections are included, correct and available for ordering.	Effectiveness: <ul style="list-style-type: none"> • Are the selections easy to access? • Easy to read? Efficiency: <ul style="list-style-type: none"> • Is there a real-time feedback to show the user has made a selection?
EC02	(FR02) The Customer's App allows for quantities to be entered of order products. Does the App allow customers to enter a quantity for each product choice?	Client to check if quantity is available for input in the interface. During Desk Checking and Final System Checking Quantity is able to be and is correctly calculated.	Effectiveness: <ul style="list-style-type: none"> • Easy to find enter object? • Reads in quantity correctly? Efficiency: <ul style="list-style-type: none"> • Is the method of data entry quick and easy to use (pull down menu rather than typing in)?

Design Review Questions

- Which of the following design tools would be most appropriate to use to identify the variables used in a solution?
 - Mock Up
 - Data Dictionary
 - Object Description Table
 - Pseudocode
- What is the difference between an event and method when describing an object?
 - An event changes an object's properties, while a method object triggers code.
 - An event triggers an object's code while a method changes an object's properties.
- Using the features of pseudocode create an algorithm for Tariq's application.
- Write THREE Evaluation Criteria for the Canteen App.

Development

The development stage involves the manipulation of data to produce a solution, validation of data being input, testing whether the solution does what is intended and writing documentation.

Manipulation of data

This is the building stage of development. The design is used to create the intended solution. Depending on the solution being created, the manipulation stage should involve:

- Creating data stores
- Creating code to manipulate the data.

Validation

Validation checks that the data being entered is reasonable. There are three types of electronic validation.

- Existence Check – Tests the input object for data
- Data Type Check – Tests the data for the correct data type
- Range Check – Tests the entered data is within a range

Testing

Testing checks whether the solution does what it is intended to do. This should be done while each part of the solution is being created as well as once the entire solution has been created and is working as a whole.

Testing involves:

- Determining what needs to be tested.
- Determining what test data needs to be used for the tests.
- Determining the results that are expected for each test.
- Conducting the tests.
- Noting the actual results that occur.
- Fixing any errors that have been identified (that is, when the actual results differ from the expected results).

Documentation

Documenting involves creating user documentation that helps the users of the solution. Documentation could include:

- Internal documentation.
- User support documentation.

Review Questions

1 Which part of the development stage involves checking the data that is entered into the system?

- A. Manipulation
- B. Validation
- C. Testing
- D. Documentation

2. In the testing phase of development, how can the tester tell that a fix is required?

- A. There is no test data
- B. Expected results and actual results are the same
- C. Expected results and actual results are different
- D. All tests are run successfully

3. Provide an example in Tariq's software solution where there will be a range check.

4. Identify one example of documentation that could be developed for Tariq's app.

Evaluation

This stage involves determining how successful the project was. It is usually conducted 6-12 months after implementation so that the users have had a chance to familiarise themselves with the new system. Evaluation criteria should be written in the Design stage in order to determine whether the solution has solved the original problem. It involves a strategy to carry out the evaluation and a report outlining how well the solution fulfilled the requirements established in the analysis stage.

Determining a strategy

A strategy is needed to help find out whether the solution meets the requirements of the solution and whether the information problem was solved. This involves:

- Creating a timeline.
- Determining the data that will need to be collected.
- Determining the methods that will be used to collect the data.
- Relating data to the evaluation criteria that was created in the design stage.

Reporting

This involves assessing and documenting the extent to which the solution meets the user requirements. It reports what worked well and the requirements that were not met. Ultimately, this involves reporting whether the information problem was solved.

Review Questions

1. When should the evaluation occur?

- A. Before the information problem is noticed
- B. During the design stage
- C. When the solution is implemented
- D. After the solution has been used for a period of time

2. What is the purpose of evaluation?

- A. Determine whether the solution has solved the information problem
- B. Determine the cause of the information problem
- C. Determine the data that will need to be collected
- D. Determine who the users of the system are

3. Describe a strategy you could use to evaluate Tariq's application and give a justification.

Solutions to Review Questions

Analysis

1. The correct answer is: C

An interview with Tariq as the client and the use of a survey can provide information from students.

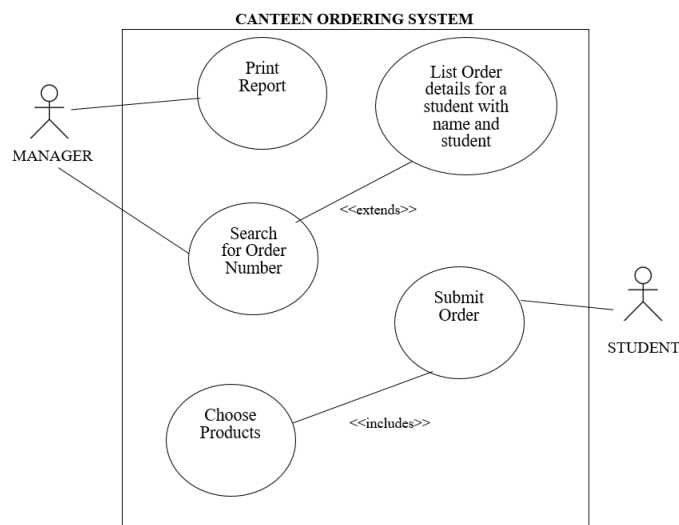
2. The correct answer is: C

Functional requirements focus on what the solution needs to do (C). The other options refer to non-functional requirements: user friendliness (A), maintainability (B) and portability (D).

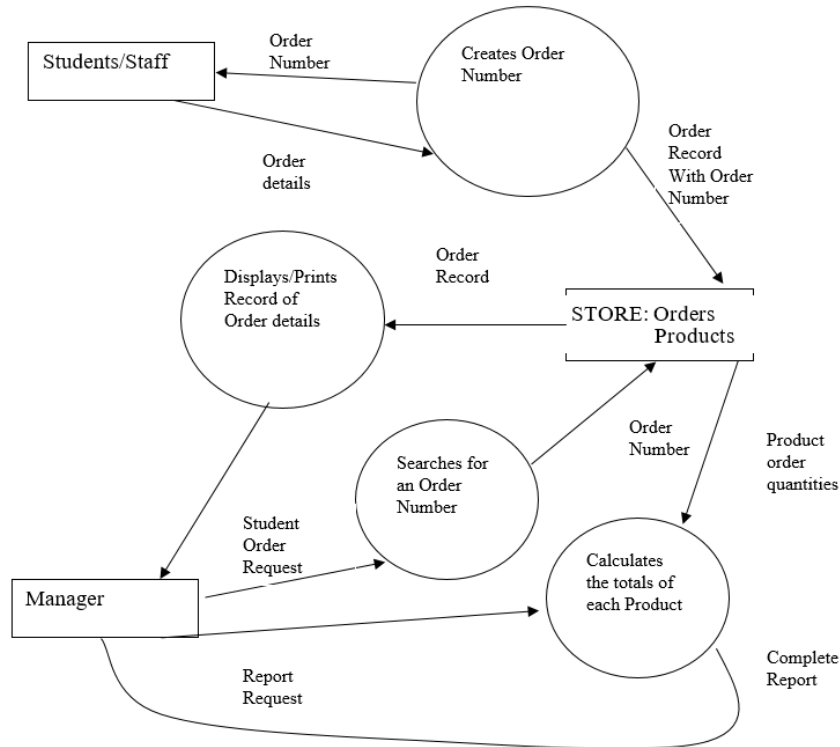
3. The correct answer is: D

Constraints are the conditions to be considered when creating the solution. The constraints that are listed refer to cost (A), level of expertise (B) and availability of equipment (C). D is something that needs to be taken into account during the project, however, it does not place a restriction as to the solution that is being developed.

4. Use Case



5. Data Flow Diagram



Design

1. B. The most appropriate tool to use to identify the variables used in a solution is a Data Dictionary.
2. B. An event object triggers code while a method object changes one of its properties.

3. Canteen Application

```

START
  Read in "Order my Lunch"
  Open "Type of Food" Form
  Read in "Type"

  If Type = Hot Food Then
    Open "Hot Food" form
  ElseIf Type = Hot Pastry Then
    Open "Hot Pastry" form
  ElseIf Type = Sandwiches Then
    Open "Sandwiches" form
  ElseIf Type = Sushi Then
    Open "Sushi" form
  EndIF
END
  
```

```

START
  Form "Hot Food"
  Read in Checkboxes
  CASE:
    Pasta Price ← $5
    Rice Price ← $4
    Curry Price ← $4.50
  EndCASE
  Read in Amount
  Total ← Price * Amount
  Display: Total, Hot Food
  Type
END
  
```

```

START
  Form "Sandwiches"
  Read in Checkboxes
  CASE:
    Chicken ← $5
    Ham ← $4
    Cheese ← $4.50
  EndCASE
  Read in Amount
  Total ← Price * Amount
  Display: Total, Sandwich
  Type
END
  
```

4.

- a) Does the application allow the user to enter lunch item sections?
- b) Is the list of lunch orders correct?
- c) 1. Does the app create a record with correct data? 2. Is the record stored in the database? 3. Is the report containing all the records correct?

Development

1 B. Validation

2. C. Expected results and actual results are different

3. The Student ID could be testing for a range check to see if it has been entered correctly.

Quantity could be tested with a range check to ensure no order has accidentally ordered more than 3 of any item.

4. The code can include comments to make sure it is easy to read.

A user manual on how to use the software can be prepared for Tariq.

Evaluation

1. The correct answer is: D

Evaluation cannot occur before the problem is noticed (A). In the design stage (B) the criteria is determined, but the solution isn't actually evaluated at this time. If evaluation is to occur when the system is implemented (C), it will not give a full picture regarding whether the problem has been solved. Evaluating after the system has been used for a while will help ensure that all aspects of the system have been used, which will help determine if it meets the required needs.

2. The correct answer is: A

Evaluation is required to help determine whether the information problem has been solved (A). The cause of the information problem (B) and the users of the system (D) should be established in the analysis stage. The data to be collected (C) is a strategy used in the evaluation stage.

3. Strategies include: Observing Students using the application; Reading all error reports; Interviewing Tariq about how the system works and meets his needs.