ANALYSIS (9 marks)

The analysis of the current system needs to be:

1. Fully or nearly fully scoped analysis of a real problem, presented in a way **that a third party can understand**.
2. Requirements fully documented in a set of measurable and appropriate specific objectives, covering all required functionality of the solution or areas of investigation.
3. Requirements arrived at by considering, through dialogue, the needs of the intended users of the system, or recipients of the outcomes for investigative projects.
4. Problem sufficiently well modelled to be of use in subsequent stages.

You should explain to a third party (i.e. the examiner) that you have identified a problem which a real end-user wants to solve using a computer system. OR an topic that can be investigated through the creation of a computer system.

To do this effectively, you need to be able to ask the user questions about the current ways in which their current system operates, so that you can find ways to improve it. This is why it is important to find an end-user who is able to communicate and provide feedback as you progress through the project.

The analysis should contain enough information so that the design could be implemented by the third party. You need to use a formal method of finding information about the organisation and document this in a thorough manner.

Gaining the skill of thinking ahead

An important skill to gain is that of “thinking ahead”. This is where you look ahead at the programming requirements and plan for the time that will be needed. You should think ahead so much that you can understand and remember the actual outline of the system, the subroutines to be used and the users main objectives. Thinking ahead also allows you to identify the critical path of the project so that you can **complete the most important part first**.

Section summary

This analysis section is divided into the following sub-sections:

- Identification of the problem. This is where you find out about the problem from the end-user.
- The current system and methods used. From your conversations with the end-user, you should be able to provide a full description of the current system that exists.
- Identification of objectives. You should then provide a list of objectives in a format that can be referred to later in the Evaluation of the project.
- System model. This final part provides an outline data model of the system. If necessary, you should also provide a list of algorithms needed, a system flowchart showing the top level of the proposed system.

Elements that GOOD Analysis have (End User):

Clear description of the End Users and their Problems

The first stage of your project is to explain the key problem areas, as well as provide an outline of the organisation and user where your project is based

1. Problem areas

The problem areas that you write about should look at the business and/or organisational areas that you are focusing on and the ways in which the use of a computer system would improve the way in which work is completed. You should write an initial list of problem areas you are seeking to solve
It is also important to clearly identify a real end-user as this makes your project more valid.

2. Follow these steps to do this:

(a) Write a short paragraph about your end-user as follows:

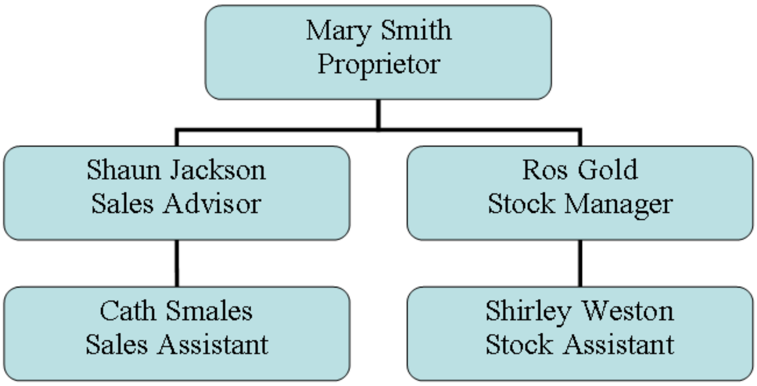
- Introduce your end-user(s) who will be your main contact throughout the time that you are completing the project.
- Make sure you state the full name of your end-user and their title/job in the organisation.

(b) Write some information about the organisation as follows:

- Write the name of the organisation that the end-user works for and their address.
- Write a description of what the organisation does and the locality in which it operates.
- Write the approximate number of employees that the organisation has, together with an estimate of the number of customers and suppliers it has.

(c) Draw a diagram showing the structure of the organisation in terms of the roles and tasks of each individual.

- This is best done using a hierarchical diagram with the names of the people in the organisation, as well as their roles and line management.
- This should be what you understand the roles and structure to be (which may not necessarily be completely accurate) and in which you can suggest changes.
- An example of an organisation diagram is shown here:



(d) Draw a hierarchy chart showing the problem areas and the names of the people in your organisation who work in those areas.

(e) Finish off this section with a brief summary of the key points that you have written.

3. Activity: Process allocation To understand the background to the problem, it is often useful to delve into more detail about the way in which the manual or computerised systems in the organisation work, as well as the organisation itself. Follow these steps to do this: • Write a paragraph (or draw a table) showing the problem areas and the names of the people in your organisation who work in those areas. • Write a brief summary of the key points and explain why you have chosen your specific problem area to work on.	❑	
4. Activity: The nature of the manual processes Most organisations have manual processes that are used on a day-to-day basis. For your specific organisation, list four manual processes Remember that, at this stage, you are writing only your initial impressions and how you think the system works		
5. Activity: Identification of the problem For the area that you have chosen, try to draw both the manual and computerised processes of a flowchart showing how you think the current system works.		
▪ To get a list of problem areas, look at the general areas that a business or organisation must have in order to function. These could include: marketing, sales, stock control, accounting, payroll and production. Try to adapt these to fit your organisation. ▪ The aim of your project is to implement a system for the end- user to improve the systems that already exist. Try to write how you think a computerised system could do this. Flowcharts are useful diagrammatical aids to show how processes work both in a manual system and a computerised system. When you draw your flowcharts, try to use flowcharting software and keep to the standard symbols for inputs, processes, storage and outputs		

The current system and methods used

You should now begin to use the preliminary work that you have done in your Project Proposal Form. Much of this section requires:

- A general explanation of the areas of the organisation that you think is in need of computerisation.
- Your opinion of what needs to be done in the organisation.
- Your ideas for this section from informal discussions with your end-user.

Remember that you should not try to put too much detail as you will elaborate on it later on.

Existing manual and computer-based methods

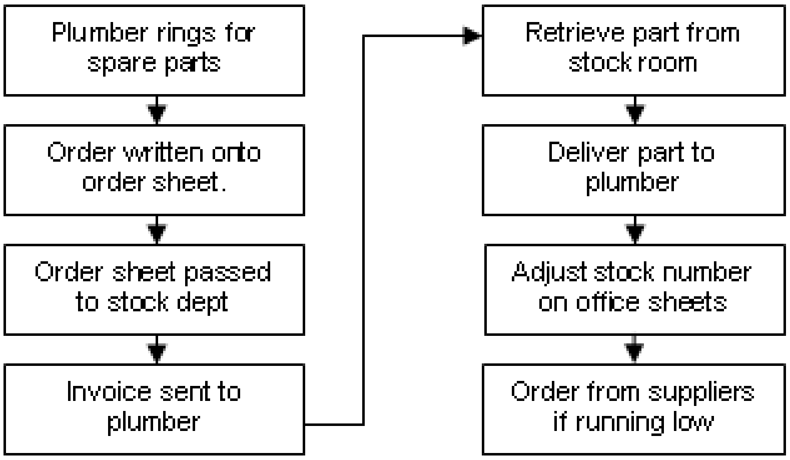
Guidance	Done	Page No
8. Description of existing manual methods	□	……

Some organisations will already have a manual system that they are using successfully, and would like you to computerise it.

If this is the case, you should complete the following:

- Describe in general the type of system they have. You can use diagrams, e.g. system flowcharts, or write a text description.

  Below is an example of a flowchart for a plumbing company:


Plumber rings for spare parts
Order written onto order sheet.
Order sheet passed to stock dept
Invoice sent to plumber
Retrieve part from stock room
Deliver part to plumber
Adjust stock number on office sheets
Order from suppliers if running low

Note: The description you write could be based on textbook research about such systems, or by your own thoughts about how such a system operates.

- Describe your initial impression of these manual methods in terms of their efficiency and the data that is inputted, processed, outputted and stored.

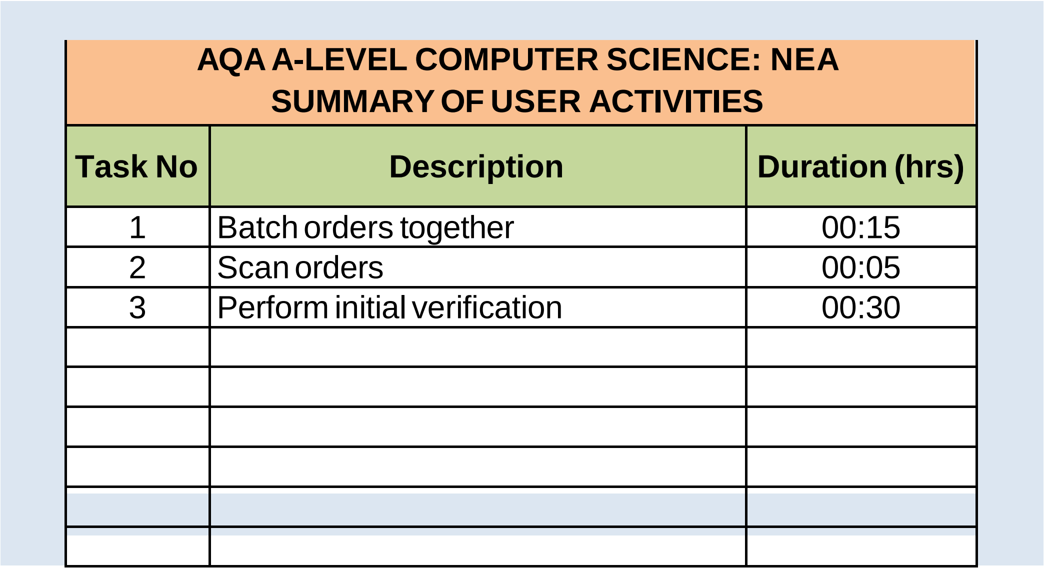
  For example: if your end-user has a manual stock keeping system, you need to research or interpret how such a manual system would work.

- Try to include a table showing your understanding of the inputs, processes, outputs and storage in your description of the project area.

9. Description of existing computer-based methods ❑ ……
   In the same way that you described the current manual methods if the organisation uses a lot of manual systems, you should do the same for any computerised systems that are used in the organisation.

10. Summary list of user activities ❑ ……
    To summarise your work, complete a table showing a list of user activities. Try to list the tasks that the user does in your organisation, as well as an estimate of the time it takes to complete these activities.

    An example of a typical table is shown below:



AQA A-LEVEL COMPUTER SCIENCE: NEA
SUMMARY OF USER ACTIVITIES

Task No	Description	Duration (hrs)
1	Batch orders together	00:15
2	Scan orders	00:05
3	Perform initial verification	00:30

Remember the following:

- Much of the information that you write in this section is dependent on what you think should be happening for the particular systems and processes in your organisation.

- When you perform your formal investigation with the end-user, you may have to alter these initial ideas.

✍ **High Mark Tips**	🕮 **Project Example**
▪ It is important to remember that this section is your initial impression or understanding of how the system should work. It may not be exactly how it works, and you'll find out more about this in the next section. In order to understand how, for example, a general booking system should work, you'll need to research such a system yourself. From knowing the general principles of the system, you'll be better prepared to collect information and ask insightful and probing questions about your end-user's system.	From my preliminary work with the company, I can see that the following are the current manual methods used: allocation of the staff to their duties, stock control of the parts sold to plumbers and keeping records that are needed for tax and VAT purposes. The current system of manual filing is prone to loss of records, so a computer system that is properly backed-up and has passwords allocated for sensitive data would be an improvement. ASC Distribution does not currently have computers, so the proposed system would be the first time that they have used a computer. I have included a flowchart of their manual processes to show my understanding of their current system.

Identification of objectives

The end-user is the key to your project. In this section, you will need to interrogate the end-user to find out exactly what their requirements of the computer system are. These requirements are known as the system objectives.

The objectives should confirm your initial work in the identification of the problem. If their requirements do not confirm your initial ideas, you should write a paragraph explaining what needs changing and state that this change has been made as a result of discussions with your end-user.

Identification of objectives

Guidance	Done	Page No
11. User objectives and aims	□	……

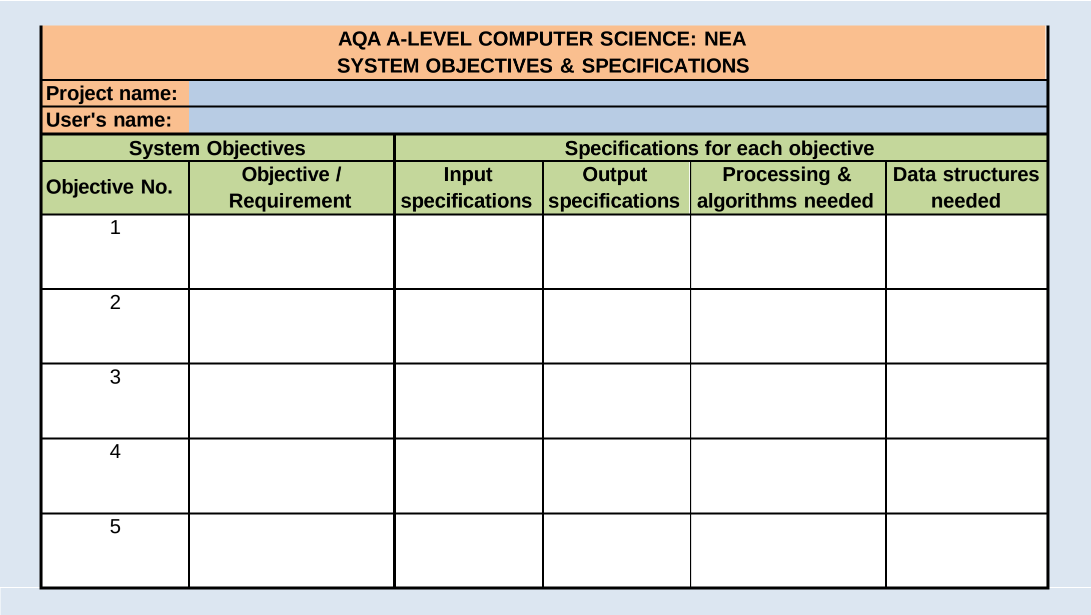
You should now be able to consult with your end-user to agree a set of general objectives of the system and more specific aims or specifications relating to the processing tasks.

Follow these general points when writing the aims and objectives:

- The objectives should be concerned with how the system should work overall, as well as time and cost constraints.
- The aim/specification for each objective should look at each particular part of the system you are going to implement, and outline what the user's aims are in terms of each of the following:
  - The input data expected to be entered into the system.
  - The results and output data that the system is expected to produce.
- You should then be able to write the processing algorithms needed to convert the input data into the output results.
- While you are thinking about the algorithms, you should also be able to decide which data structure is best for the processing to take place.

- Examples of the choice of data structure are:
    - A database for large volumes of data.
    - A queue to model a real-life queue.
    - A stack to model data that is generated sequentially and processed as it is generated.
    - A list to model data stored sequentially.
    - A linked list to model random data that is connected.
    - A tree to model inter-relationships between individual data items.

Use the table below to help in organising your aims and objectives:



AQA A-LEVEL COMPUTER SCIENCE: NEA SYSTEM OBJECTIVES & SPECIFICATIONS					
Project name:					
User's name:					
System Objectives		Specifications for each objective			
Objective No.	Objective / Requirement	Input specifications	Output specifications	Processing & algorithms needed	Data structures needed
1					
2					
3					
4					
5					

12. Processing requirements ❑ ……

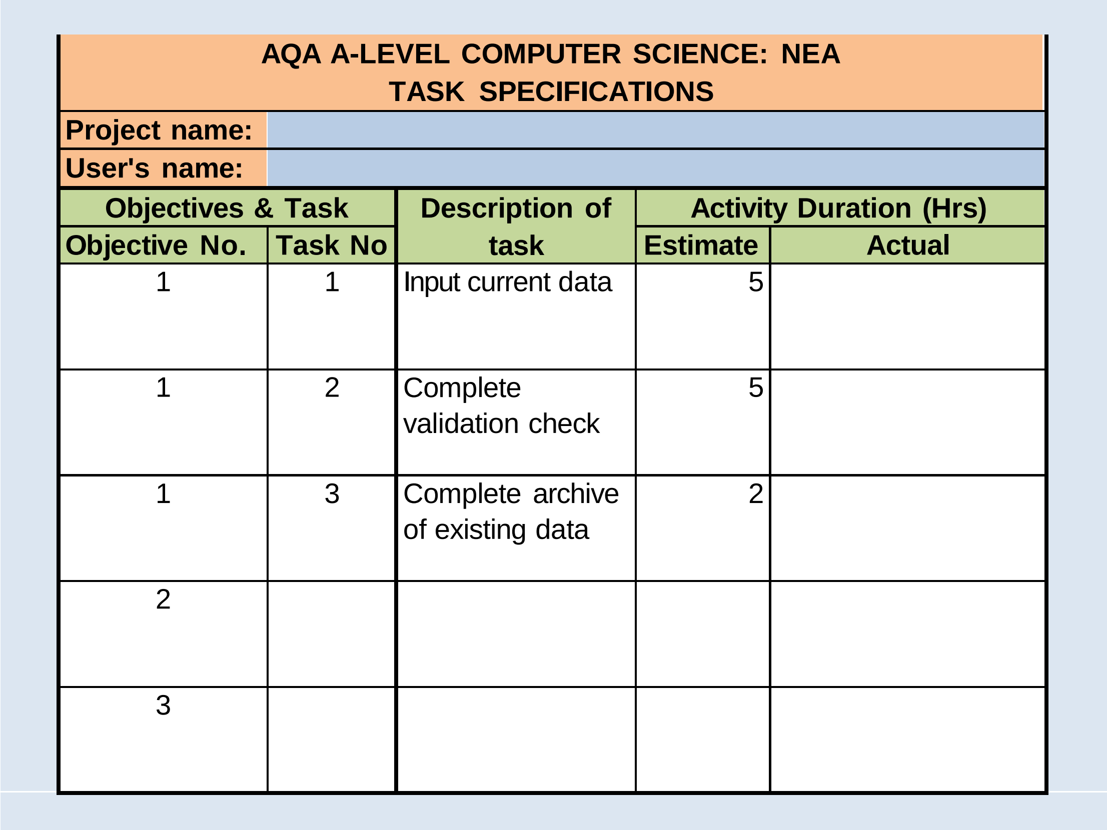
Processing is an important part of a computer system. As with the general system objectives and specifications, write a list of general processing requirements that the end-user will need.

You can either use the table above, or create a separate table.

13. <u>Activity: List of tasks needed by the users</u> □ ……

Each objective that you have identified now needs to be broken down into a series of sequential tasks.

Here is an example table of a task list that you can use to help you get started.



AQA A-LEVEL COMPUTER SCIENCE: NEA TASK SPECIFICATIONS				
Project name:				
User's name:				
Objectives & Task		**Description of task**	**Activity Duration (Hrs)**	
Objective No.	**Task No**		**Estimate**	**Actual**
1	1	Input current data	5	
1	2	Complete validation check	5	
1	3	Complete archive of existing data	2	
2				
3				

Follow these steps to complete this task list:

- Look back at your list of user aims and objectives and break each of these down into tasks. Each task must be something that can be ticked-off when it has been achieved.

- Because you are not aiming to show too much detail at this stage of your work when completing the task list, try to sub-divide each objective into a maximum of five tasks.

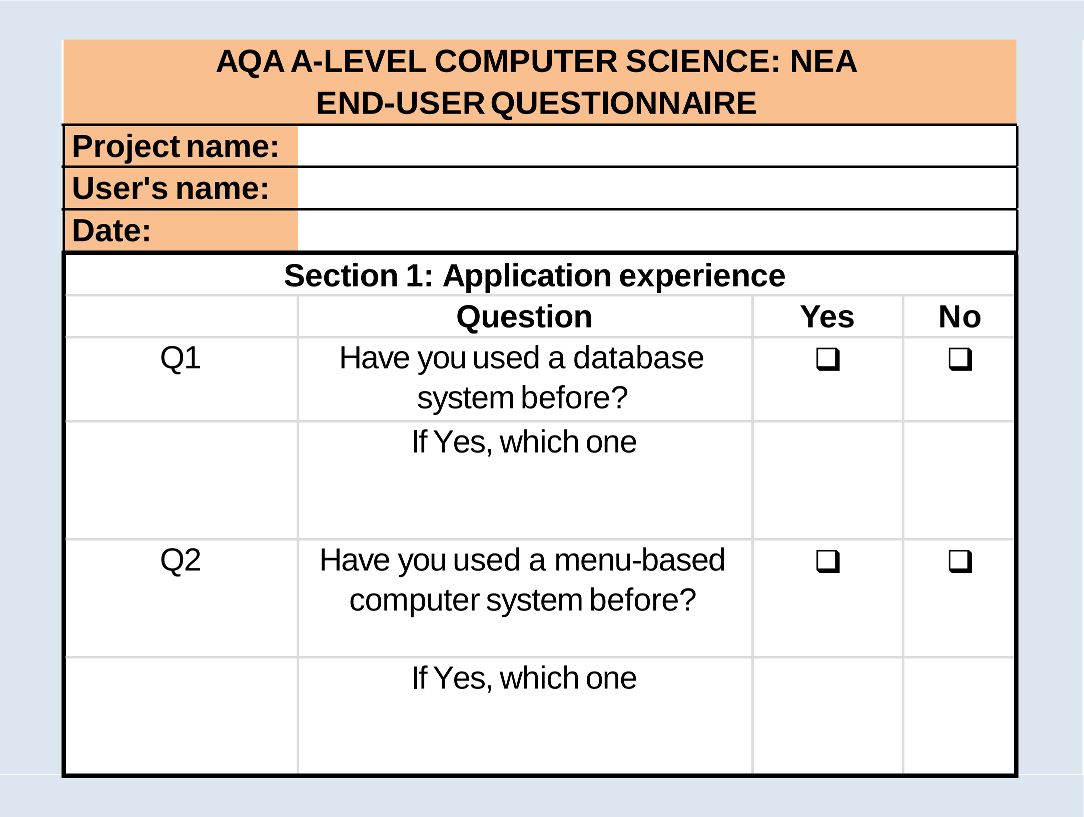
- Write the task description that includes one or some of the following: the input, processing, algorithms to be used, output or storage that takes place.

- Write an estimate of the number of hours that the task will take.

- The actual activity hours will be completed later on in your evaluation.

14. User’s computer skills □ ……
This is an important factor to take into account. It will determine how you design the user interface of the system and how you write the user manual. A good way of establishing these is by getting the end-user to complete a separate skills questionnaire. This shows that you have completed a dialogue with the end-user.

Below is an example to help you write your own:



AQA A-LEVEL COMPUTER SCIENCE: NEA
END-USER QUESTIONNAIRE

Project name:	
User's name:	
Date:	

Section 1: Application experience

	Question	**Yes**	**No**
Q1	Have you used a database system before?	□	□
	If Yes, which one		
Q2	Have you used a menu-based computer system before?	□	□
	If Yes, which one		

15. Exercise: □ ……
From your dialogue with the end-user, you will have gained an idea of what hardware and software is available in the organisation.

Complete this exercise to investigate the limitations of these.

- List these limitations in detail including the specifications of both the hardware and software.

- If the organisation already has computer equipment, you should perform a comparison between those computers and the one that you will be doing most of your development work. Try to focus on the way in which the system works. For example, here are some questions that you could ask:

  - Is there a network?

  - Is there a backup made of the data?

  - Are there any Internet links for updating software?

  - Are there sufficient printing capacities?

  - Is there a system manager that is able to install executable code onto the system?

  - Are you able to use an interpreter or compiler on the system?

- List the requirements of the system that you think may not be suitable for computerisation and state why this is so.

16. Activity: Software and hardware capabilities ❑ ……
For the software programs and programming languages being used:

- Draw a table showing the processes that need to be done and how you will perform the task using your chosen software development tool.

- The list of processes does not have to be detailed. It should come from the summary of the questionnaires, interviews and other data collection methods that you have used.

17. Activity: Hardware and software required ❑ ……
If your organisation does not have any hardware or software currently, you should suggest a list of hardware and software that is needed. Follow these guidelines for the contents of your list.

- For the hardware, produce a minimum technical specification as well as a recommended specification. This should be broken down into areas such as:

- Hard disk/storage
- Memory
- Processor
- Graphics
- Secondary devices
- Printing
- Output devices
- Input devices

Use the space below to write this:

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

- For the software, include a brief description of the features of the programming language you will be using. Complete this in the space given below:

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

High Mark Tips

- The best way to write your task descriptions is to begin each description with a verb that describes what you need to do in order to achieve the objective.

- Try to extend your task list to add the number of elapsed days that it takes for you to complete the tasks.

  Do this by adding a column under the “Estimates” and “Actual” parts with the heading “Elapsed Hrs”.

  You can work out the elapsed days with this formula.

  Elapsed Days =

  Activity Hrs
  No of working hrs per day

  The number of working hours per day should be no more than 8-10 hours.

Project Example

I have obtained a list of user objectives and completed a table showing the tasks needed by the end-user. These are shown on page 4A.

I have also produced a user skill evaluation questionnaire so that I can get an idea about how computer-literate the end users are. The results of these are shown on page 4B, together with a chart showing the average score of the users in the numerical questions.

System modelling

A good analysis of the current system should look at the sources of data and the way in which the computer system should process the data. To do this, you should show how data and processing interact.

- You will need to develop an understanding of the way in which data flows through the organisation. This should be a simple outline view of how documents (which represent the data) travel around the organisation, followed by a Level-0 Data Flow Diagram.

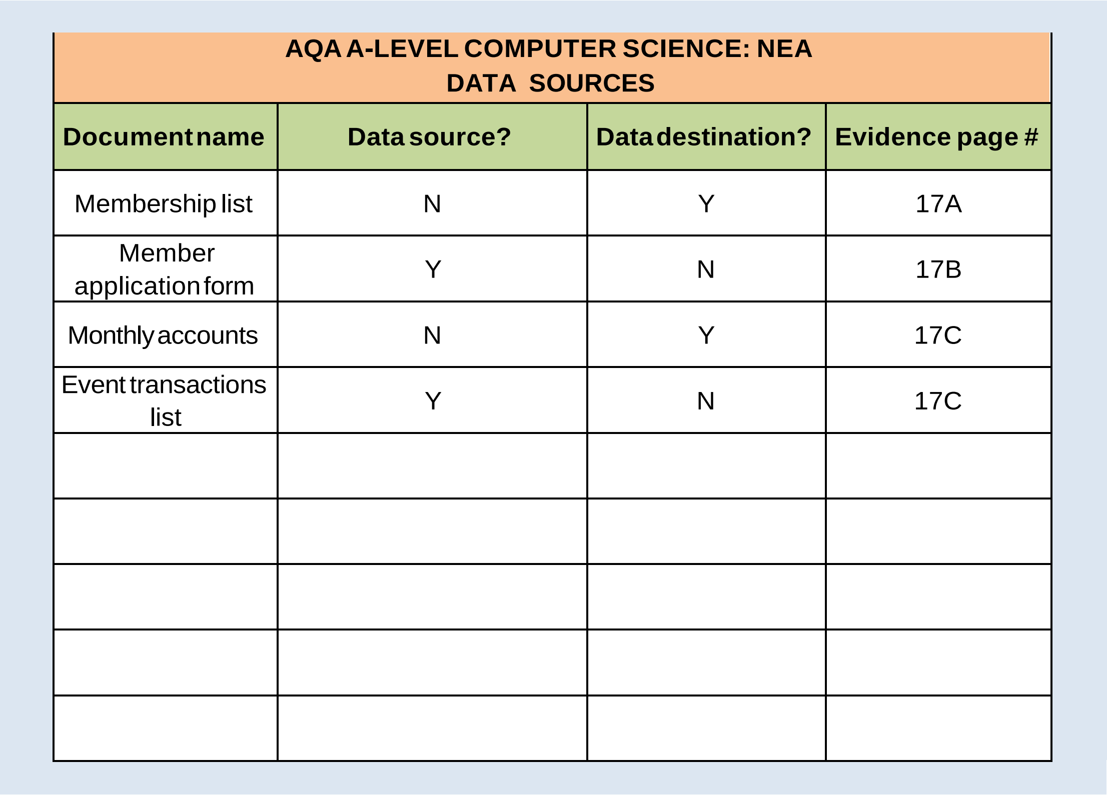
- You should also gain an understanding of the objects in the system if you are using object-oriented programming.

- Finally, you need to show an overview of the overall system with a list of algorithms and a system flowchart.

Note: It is important that you use this part of your report to thoroughly analyse the needs of the user in terms of what data will be used, and don't just report on the user's requirements in terms of data storage.

Data structures		
Guidance	Done	Page No
18. Identify the data sources and destinations A data source is an item or entity that creates data for entry into the computer. The data source generates the data which needs to be entered into the computer. A data destination (or data sink) is an item or entity to which computer reports, printouts, emails, faxes and other documents are sent. When you look for data destinations, ask about the entities who receive information from the organisation. Look at the manual or automated processes that occur and work out where the data that is entered into the system comes from and where it is sent. Create a table of these in your report with an indication of the actual data that comes from the source. Typical examples of data sources are: customers, suppliers, subscribers and staff.	❑	……

An example table is shown here:



AQA A-LEVEL COMPUTER SCIENCE: NEA
DATA SOURCES

Document name	Data source?	Data destination?	Evidence page #
Membership list	N	Y	17A
Member application form	Y	N	17B
Monthly accounts	N	Y	17C
Event transactions list	Y	N	17C

19. Forms used to collect data □ ……

You should have collected the actual forms used in your organisation. The forms can be used as evidence of the data sources and destinations.

Add an extra column in your list to cross-reference these forms by page number. Examples of forms that you can try to obtain copies of include:

- Requisition forms
- Order forms
- Delivery notes
- Invoices
- Stock control forms
- Job application forms

20. Data structure and size estimation □ ……

Write a list of the data structures that you propose to use in your project.

Look at the data structures and analyse any limitations that are placed on them.

One of these limitations is the volume of data that can be stored. To show this, complete a table showing the estimated size of storage space that will be required.
An example of such a table is shown below:

ESTIMATE OF STORAGE SPACE		
Table	**Field**	**Size (bytes)**
StockDetails	StockID	4
	StockName	30
	SupplierID	4
	Total	**38**

✍ High Mark Tips

- Remember that data entered into a computer system must be validated and verified, otherwise it may be inaccurate.
- An entity can be an individual, another organisation that the company is linked to, or another computer system. Be careful of including copies of forms with data about living individuals (which could be contravening the Data Protection Act) or which contain business-sensitive information. It is safest to blank out any data on forms that you are concerned about, but to leave the field names/column headings.

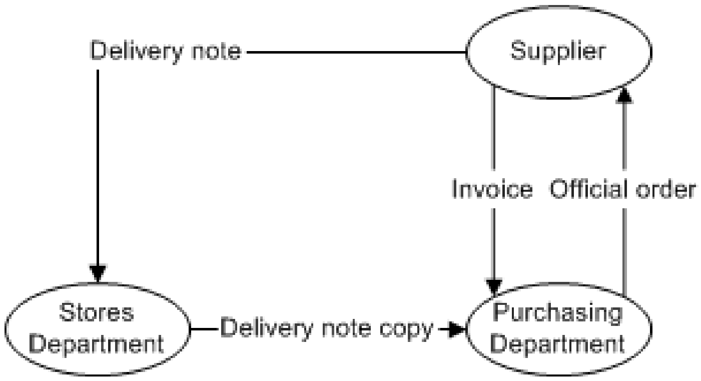
🕮 Project Example

The company has several sources and destinations of data, which are summarised in the Data Sources and Destinations table shown on page 6A. Included in the table are the page numbers of the forms obtained showing the data that is associated with each source and destination.

Data modelling with data flow diagrams

Guidance	Done	Page No
21. Document-Agency Flow Diagrams As part of the process of examining the way in which data flows in an organisation, you should try to look at the documents, forms and reports that are passed around the organisation. This information should come from your questioning of the end-user and can be summarised in a document-agency flow diagram, with the arrows representing the documents and the ovals representing the agencies. A simple example of such a diagram for a purchasing system is shown here.	❑	……

Purchasing Department Document Flow

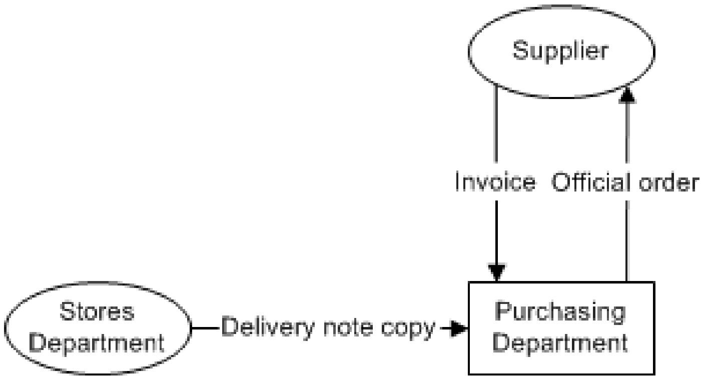


Try to draw as many of these diagrams as you can for the various parts of the organisation that you are attempting to implement.

22. Context Diagrams ❑ ……

The next stage from the Document-Agency Flow Diagrams is Context Diagrams. This is simply the document-agency diagram which focuses on a specific department or section of the organisation. It ignores document flows that take place between other departments in the organisation. In a Context Diagram, a rectangular box (instead of an oval) is used to represent the part of the organisation being looked at.

Purchasing Department Context Diagram

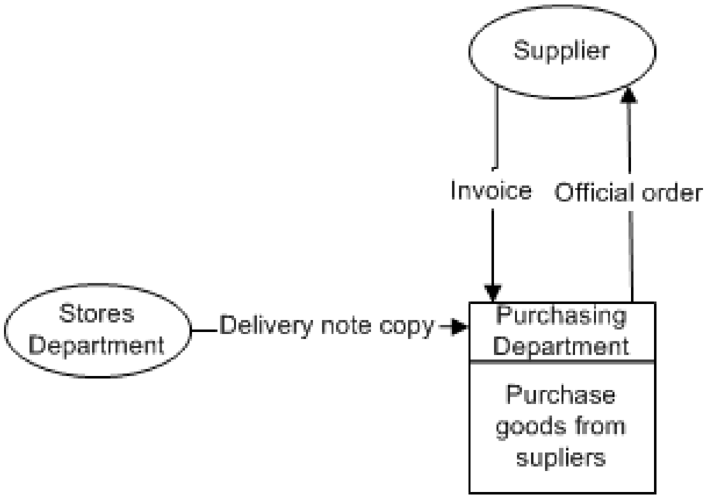


23. Level-0 Data Flow Diagrams: Existing system ❑ ……

A Data Flow Diagram (DFD) is used to show how data flows between agencies (departments, people or sections) as well as the processes that cause the data to pass between the agencies. A Level-0 DFD shows how the organisation works physically at the moment. To convert your Context Diagrams to Level-0 DFDs, just add a description of the process for the main context (the rectangle).

Below is an example:

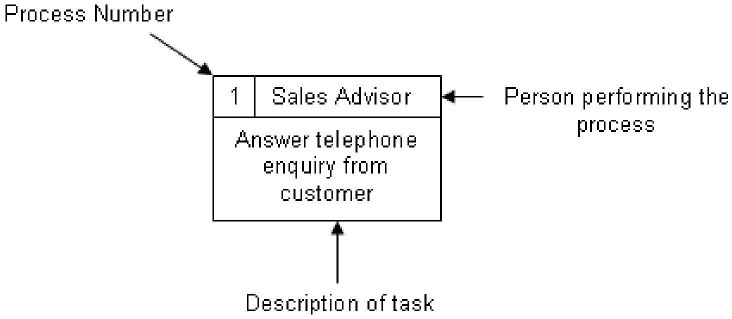
Purchasing Department Level-0 Data Flow Diagram



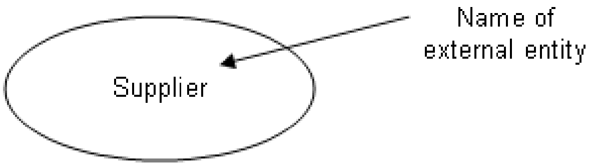
24. Activity: Draw Level-0 Data Flow Diagrams ❑ ……
Now draw data flow diagrams of the existing systems in your organisation by looking back at the examples given.

25. Level-1 Data Flow Diagrams ❑ ……
To further analyse the way in which data moves in the organisation, you need to look at greater depth and produce Level-1 DFDs. Follow these steps:

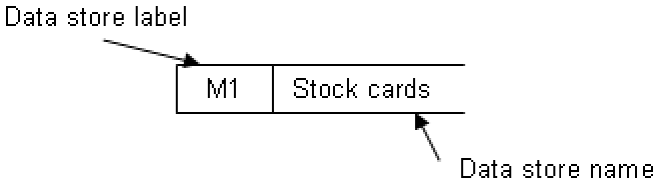
- For each of your Level-0 DFDs, break down the main process into a sequence of processes with the job title of the person who performs the process indicated. Unless you have been taught otherwise, use the following convention for showing your processes:


Process Number
1
Sales Advisor
Person performing the process
Answer telephone enquiry from customer
Description of task

- For each of these processes, write a list of external entities that has an impact on the process. Then draw an arrow with a label to show which document/data is transferred. Be careful to get the data travelling in the correct direction. Unless you have been taught otherwise, use an oval shape for the external entity with the name of the entity in the centre.



- Now list any manual or computer-based names of the documents/files where the data is stored from each of the processes. If the data store is a manual one, label it with the letter M followed by a sequential number, otherwise use the letter D.



Add the data items that are being transferred in the same way as you did for external entities.

26. Complete your work ❑ ……

To complete your Level-1 DFDs, copy all your processes, external entities and data stores into a complete diagram showing all the data flows. If you think that a process cannot be simplified further, then the convention is to place an asterisk (*) next to its name.

Note: You should produce your Level-1 DFDs for the existing and proposed system.

High Mark Tips

- When creating Data Flow Diagrams, it is best to first concentrate on the processes that cause the data to flow. After this, identify the data which flows, and finally where the data is stored or the external entity where it is sent to. Get into the habit of drawing your processes down the middle of the page, the external entities to the left and the data stores to the right of the page.

- Remember that Level-0 DFDs do not contain anything about storage of data, and usually contain only one central process. When completing your Level-0 DFDs, try to place the external entities to the central department or section to the left of diagram. This will help you later on when you produce more detailed diagrams.

- You may want to write a short paragraph to comment on the problems with the systems that you can see from the diagrams that you have drawn.

Project Example

To show how the data moves in the current system, I have completed a Level-1 Data Flow Diagram.

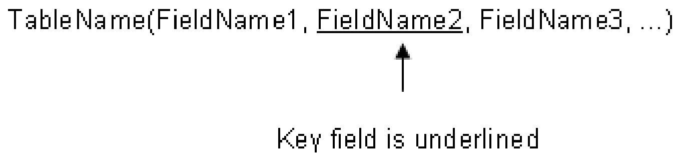
In order to do this, I looked at the documents that are used in the booking system, the MOT system and the supplier system. Three different DFDs resulted, all of which are shown on the next three pages (p15-17).

Normalisation of data

Guidance Done Page No

27. Record definitions □ ……

Look back at your data dictionary and forms which you have collected as your data sources and, for each one, write down the file/table name and its fields using the convention as shown below:



TableName(FieldName1, FieldName2, FieldName3, ...)

Remember: A key field is used to identify a specific record in the table. This means that the key field can be used to find one (and only one) record. Each table should have a key field, which is usually an ID number or a combination of numbers.

28. Normalisation – Un-normalised Form (UNF) □ ……

Now that you have identified your tables, you should re-organise them so that data duplication is removed. This will help your system to save on disk space and also ensure data integrity. Normalisation is the name that is given to this process of re-organising the tables.

The first stage of normalisation is to write down all the data that is needed for a specific document, form or table and to identify the key attribute (or field). This is known as Un-normalised Form (UNF).

29. Exercise: □ ……

- Look back at your forms, record definitions and data dictionary and complete the Normalisation Table shown below by entering the attribute names in the UNF Attributes column.

Below is a table showing fields for a stock details form.

NORMALISATION TABLE				
UNF Attributes	**UNF Level**			
StockID	1			
ItemDesc	1			
SupplierCode	2			
SupplierName	2			
SupplierAddress	2			
SupplierContact	2			
PickerName	2			
LastCount	2			

- Underline the key attribute in the list. In the example given above, the key attribute is StockID as this identifies one particular stock item.

- To fill-in the UNF Level column, start with UNF Level number 1 for the first attribute and work your way down the list of attributes. If any items are a repeating group, give them the level number 2, and so on. In the example above, for each stock item there is more than one supplier so any attributes concerning suppliers are given UNF level 2.

30. Normalisation – First Normal Form (1NF) ❑ ……

The next stage is to make the data into First Normal Form. This is done by separating the repeating data identified into its own table. Below is an example for a stock details form:

NORMALISATION TABLE				
UNF Attributes	**UNF Level**	**1NF Attributes**		
StockID	1	StockID		
ItemDesc	1	ItemDesc		
SupplierCode	2			
SupplierName	2	SupplierCode		
SupplierAddress	2	StockID		
SupplierContact	2	SupplierName		
PickerName	2	SupplierAddress		
LastCount	2	SupplierContact		
		PickerName		
		LastCount		

31. Exercise: □ ……

Follow these steps to organise your data into 1NF:

- Choose a key attribute for the repeating data. In the example above, SupplierCode is the most obvious choice.
- Combine together the UNF Level 1 attributes.
- Combine together the UNF Level 2 attributes and add the key for the repeating data as well as from the UNF Level 1 group. In the example above, all the supplier attributes are combined and the StockID attribute is added.

Note: A compound key has been created for the second set of attributes.

32. Normalisation – Second Normal Form (2NF) □ ……

To make the tables into Second Normal Form, look at the compound keys and check if the other attributes really depend on either the compound key items or just one. Below is an example:

NORMALISATION TABLE				
UNF Attributes	**UNF Level**	**1NF Attributes**	**2NF Attributes**	
StockID	1	StockID	StockID	
ItemDesc	1	ItemDesc	ItemDesc	
SupplierCode	2			
SupplierName	2	SupplierCode	SupplierCode	
SupplierAddress	2	StockID	StockID	
SupplierContact	2	SupplierName	PickerName	
PickerName	2	SupplierAddress	LastCount	
LastCount	2	SupplierContact		
		PickerName	SupplierCode	
		LastCount	SupplierName	
			SupplierAddress	
			SupplierContact	

33. Exercise: □ ……

Follow these steps to organise your data into 2NF:

- Copy any groups of attributes without a compound key into the 2NF Attributes column. In the example above, the StockID group has been copied across into the 2NF Attributes column.
- For the compound key groups, go through each attribute that is not a key and check if they are determined by both

parts of the compound key or just one.

- Separate out the attributes which depend on only one part of the compound key into a separate table with the key field from the compound key. In the example above, three attributes depend only on SupplierCode and not on StockID, so they are separated out.

34. Normalisation – Third Normal Form (3NF) □ ……
To get your data into Third Normal Form, look at all the attributes and double-check to see if they are really dependent on their key(s). Below is an example:

NORMALISATION TABLE				
UNF Attributes	**UNF Level**	**1NF Attributes**	**2NF Attributes**	**3NF Attributes**
StockID	1	StockID	StockID	StockID
ItemDesc	1	ItemDesc	ItemDesc	ItemDesc
SupplierCode	2			
SupplierName	2	SupplierCode	SupplierCode	PickerID
SupplierAddress	2	StockID	StockID	PickerName
SupplierContact	2	SupplierName	PickerName	
PickerName	2	SupplierAddress	LastCount	SupplierCode
LastCount	2	SupplierContact		StockID
		PickerName	SupplierCode	PickerID*
		LastCount	SupplierName	LastCount
			SupplierAddress	
			SupplierContact	SupplierCode
				SupplierName
				SupplierAddress
				SupplierContact

35. Exercise: □ ……
Follow these steps to organise your data into 3NF:

- Work through all the 2NF attributes and check if they depend on the key. In this example, PickerName should depend on an ID field, so this has been separated out.

- Any keys of the groups of attributes taken out are left as foreign keys and marked with a star.

Copy across all the other groups of attributes into the 3NF Attributes column.

Object-oriented programming

Guidance Done Page No

Note: You should only perform an object analysis if you are producing an object-oriented programming project.

36. Activity: Identification of objects ❑ ……

If you intend to complete the project using an Object-Oriented Programming Language, you should write down a list of objects that will be created.

Remember that, for the purposes of this project, an object can be thought of as an item in an entity.

37. Activity: Class definitions ❑ ……

The next stage is to define the classes that will be used. Object classes can be thought of as being the same as entities.

The example below shows the objects for the object class Supplier.

OBJECT CLASS DIAGRAM

Supplier
SupplierCode SupplierName SupplierAddress SupplierContact

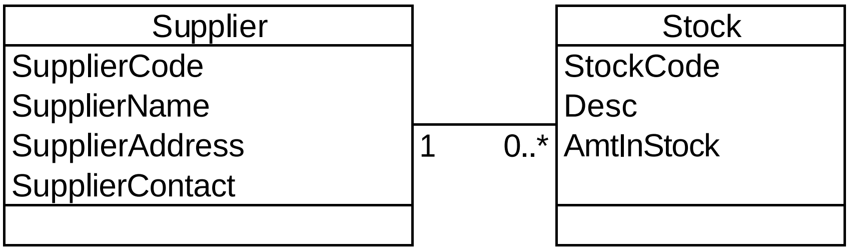
For each of your classes, you should include the class attributes (data types).

38. Activity: Association and class diagrams ❑ ……

An association is the equivalent in object-oriented analysis to a relationship. A class diagram is the equivalent to an E-R diagram.

The multiplicity of the relationship in the class diagram is shown by using numbers instead of “crow’s feet”.

As an example, the diagram below shows the class diagram for a supplier system:

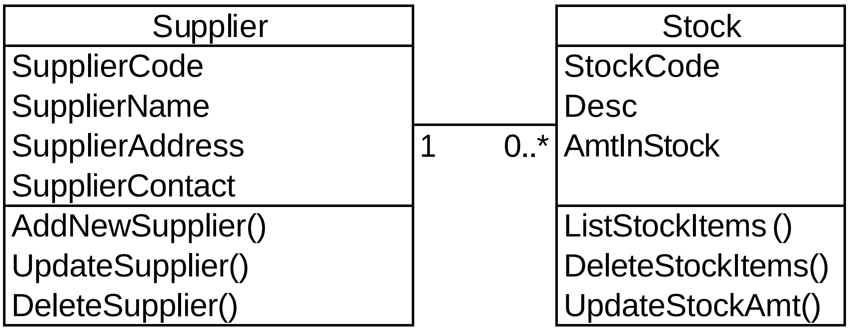


39. Activity: Encapsulation □ ……

Encapsulation is the packaging of data and operations into objects. An operation is the same as a function or procedure.

To show encapsulation in your class diagrams, you need to look at the object and the procedures or functions that act on the object.

The operations are written in the rectangular box underneath the objects. Look at the following example:



Work through your class diagrams adding encapsulation where appropriate by thinking of some initial processes that will be needed.

Prototyping and Critical Path Analysis

You should now be at the stage where you have a good understanding of the requirements of the end-user and also the individual tasks to be completed to achieve those requirements.

You now need to develop initial algorithms and a system flowchart and produce a simple prototype of the system. You also need to prioritise the work to be completed by finding the critical path.

Proposed algorithms and initial system flowcharts

Guidance	Done	Page No
40. Activity: Listing your proposed algorithms Look back at the whole of the analysis you have performed and produce a list of algorithms that are needed to complete the processing requirements.	☐	……
41. Activity: Initial System flowchart Using a system flowchart drawing tool, produce a system flowchart for the whole system.	☐	……

Teacher's Comments

○ Max. Mark

○ Mark Given

Producing a prototype

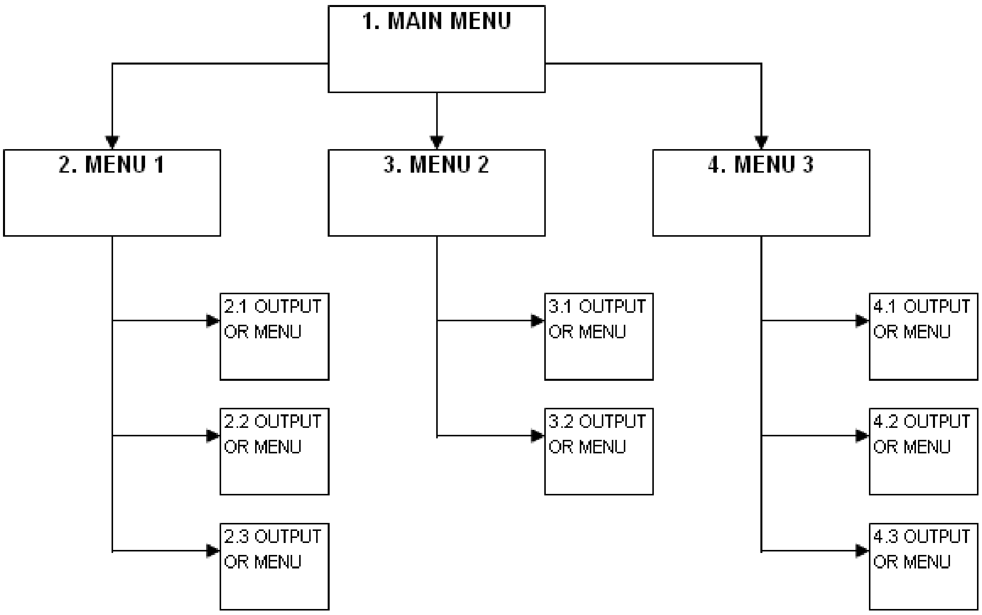
Guidance Done Page No

42. Prototyping ❑ ……

A prototype is a design of the user interface, without the processing parts of the project. To produce a prototype, you will need to produce an initial design of the menus with the different options that the user requires.

Menu designs are also called storyboards. They are used to show what forms and menus are needed and how these are linked together. An example is shown below:

HIERARCHICAL STORYBOARD



43. Activity: Drawing a menu design ❑ ……

Draw your own menu design to show how the user interface will work in your proposed system. Make sure that you include all of the features that the end-user has requested.

44. Activity: Implementing the prototype ❑ ……

Using your programming language's IDE, implement the menu design for the prototype. You will need to use the design features of the programming language to do this. Do not attach actions to any buttons.

45. Activity: Obtaining user feedback ❏ ……
Now go back to your end-user and obtain confirmation that the prototype contains all the features required and meets all the objectives of the project.

○

○

Critical path analysis

Guidance	Done	Page No
The critical path is the part of your project that everything else depends on for a working system to be obtained.		
46. Drawing a Gantt Chart	❑	……

A Gantt chart is used to show the sequence of tasks in a graphical way. To determine the critical path, you need to draw a Gantt Chart. Refer back to your table of task specifications on page 47. Work out a schedule of work so that you'll be able to complete these tasks.

Construct your Gantt Chart as follows:

- Open your spreadsheet and enter the "Time (days)" axis values. Adjust column widths so that they fit and are the same.

- Number the tasks in your table by using the format of objective number followed by a full stop, then the task number, e.g. Task 1.5. Write these numbered tasks by working down the spreadsheet.

- Start at the first task to be completed and shade cells to show the estimated amount of time to complete the task.

- Continue until you have finished all of your tasks. This is known as forward planning as you should be able to get a good idea of when in the future you'll be able to complete your project.

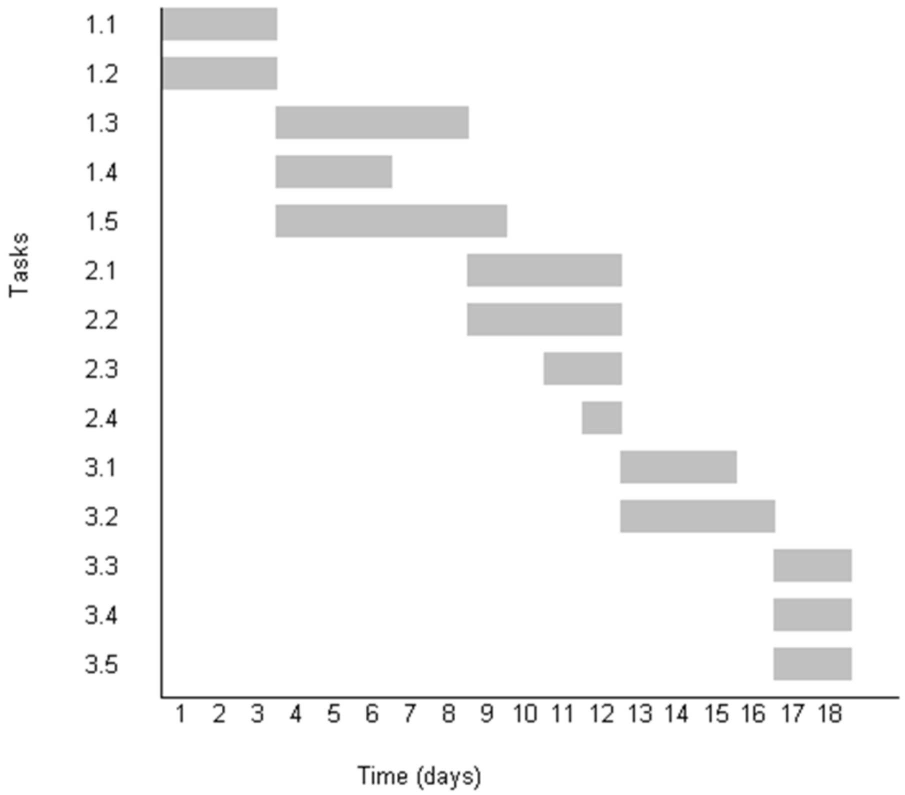
- You need to use your user tasks list to construct your Gantt chart. Look at how long the tasks will take to complete and whether a task may overlap with another task.

- You should write an explanation of why you have allocated a certain task to overlap with another task. Also use this opportunity to write when you think each objective will be achieved and when the project will be finished. Try to inform your user about the target completion dates for the objectives, and obtain confirmation that these are satisfactory.

- Ensure that the following are indicated on your Gantt Chart:
  - Progression from one task to the next.

- Predecessors to all tasks.
- Successors to all tasks.

An example Gantt Chart is shown below:



47. Obtaining the critical path ❑ ……

Using the Gantt Chart calculate the longest time taken from the start to the end of the project. This is known as the critical path.

Now highlight in red all the activities that form the critical path.

○

○