Cambridge (CIE) IGCSE ICT

Your notes

The Systems Life Cycle

Contents

Your notes

What is the system life cycle?

- The system life cycle is **a structured process** that guides the **planning**, **creation**, **testing**, and **deployment** of an information system
- Ensures systems are **systematically developed**, meeting needs of stakeholders, **minimising** risks and **maximising** efficiency
- The first stage is **analysis**, the purpose is to:
  - Gather detailed **requirements** from stakeholders and analyse them to understand **what the system should accomplish**

Research Methods

What are research methods?

- Research methods are different ways a systems analyst could **analyse the current IT system** in order to **help understand the state of the current system**
- Research methods enable an analyst to **identify areas for improvement**
- There are four main research methods:

Observation

- Watching users **interact with the current system** to see how it acts/works

Questionnaires

- A **structured set of pre-determined questions** to enable an analyst to get the views of the existing system from:
  - **Workforce**
  - **Clients**
  - **Other system users**

Interviews

- **One-to-one question and answer sessions** to enable an analyst to dig deeper in to the views of existing users of the system

Looking at existing system (including paperwork)

- A full look at the **state of the existing system** including:
  - **Checking training manuals**
  - **How paperwork is filed**
  - **Operational instructions**

- **Accounts etc.**
- Helps an analyst to **identify the scale of the a problem**, **memory requirements** and any **input/output devices**

Research method	Advantages	Disadvantages
Observation	▪ Reliable data ▪ Inexpensive ▪ Good for getting an overall picture of existing system	▪ People don't always behave in the same way under observation
Questionnaires	▪ Quick ▪ Inexpensive ▪ Can be anonymous	▪ Can't ask follow up questions ▪ Low response rate ▪ Answers might be vague
Interviews	▪ Can monitor body language and facial queues ▪ Gives employees an opportunity to express opinions in safe environment ▪ Can change questions based on responses	▪ Time consuming ▪ Cannot remain anonymous ▪ Employees may be uncomfortable and not give honest responses in fear of repercussions
Looking at existing system	▪ Obtain information that can't be obtained from other methods	▪ Time consuming which can be costly

Analyse Current System

How do you analyse the current system?

- To fully analyse the current system , there are a number of **key aspects** that must be identified, these include:
  - **Inputs**: data or information entered into the system
  - **Outputs**: data or information generated by the system
  - **Processing**: tasks performed by the system on the inputs to produce the outputs
  - **Problems**: issues that users face with the current system
  - **User requirements**: what users need from the new system

Your notes

- **Information requirements**: data or information the new system must process
- Data flow diagrams (**DFDs**) are useful for analysts to understand the inputs, outputs and processing in the existing system
- A **requirements specification** is drawn up from the results of the analysis
  - How the new system will work, be deployed and a time scale

Your notes

Hardware and software selection

- **Identify suitable hardware**
  - Consider system requirements, compatibility, and cost
  - Justify choices based on user needs and system performance
- **Identify suitable software**
  - Consider functionality, compatibility, and ease of use
  - Justify choices based on user requirements and system efficiency

Worked Example

A small company makes toys and then delivers them to shops.
Throughout the day orders are received by the company from its customers. The office workers in the finance department create and store an invoice for each order. They are too busy to be disturbed by their work.
Delivery drivers receive copies of the invoices which they will pass on to their customers. The drivers make a large number of deliveries per day and do not return to the office.

A systems analyst will research the current system and suggest improvements to be made.

For each type of employee identified above, describe the most suitable method of collecting information from them, giving a reason for your choice.

[4]

Answer

Office workers:

Observation of the processes taking place [1]
Looking at existing paperwork [1]

Reason:

One of:

Enables the systems analyst to see the whole system [1]
There are too many workers to interview them all [1]
Questionnaires/interviews would stop them from working on their tasks [1]
Can see how the files are stored/processes undertaken [1]

It allows information to be obtained that cannot be obtained in other ways [1]
enables necessary storage, and computer equipment to be identified [1]
If they are observed, then they may change the way they work [1]
They are too busy to be interviewed [1]

Delivery drivers:

Questionnaires could be handed out [1]

Reason:

One of:

They can complete them in their own time/at their leisure [1]
Questionnaires tend to be more accurate [1]
The data can be collated more quickly as everyone can complete at the [1]
same time rather than interviewing which is one after the other [1]
Individuals remain anonymous therefore they will be more truthful/reliable [1]
Easier to analyse [1]

Your notes

Your notes

What is the system life cycle?

- The system life cycle is **a structured process** that guides the **planning**, **creation**, **testing**, and **deployment** of an information system
- Ensures systems are **systematically developed**, meeting needs of stakeholders, **minimising** risks and **maximising** efficiency
- The second stage is **design**, the purpose is to:
  - **Develop** architectural **blueprints for the system**, including **database design**, **user interfaces**, and **system interfaces**.

File/Data Structures

What are file/data structures?

- File structures are considered during the design stage of the systems life cycle and consist of defining:
  - **Field names**
  - **Filed lengths**
  - **Data types**
  - **Primary keys**
- A file **consists of records** and records are **made up of fields**
- Each record is **identified by its unique primary key** field

Data types

Data type	Description	Examples
Alphanumeric	Stores a combination of characters (**letters or text**) and numeric data	Usernames, postcodes, product codes, phone numbers
Character	A **single letter/symbol**	A, B ,C etc.
Text	Stores a combination of characters (**letters, text, symbols, special characters etc**.) and numeric data	This_is_an_example
Boolean	Stores data in a **Yes/No or True/False format**	Y or N

Numeric	Stores numerical data: ▪ **Integers** ▪ **Decimals** ▪ **Currency** ▪ **Date/Time**	100 1.50 £2.99 18/05/2024 or 14:58

Your notes

Case Study

A file structure for records containing information about cars

Field name	Field length	Data type	Primary key?
Car_ID	6	Alphanumeric	Yes
Year	4	Numeric: integer	No
Make	20	Text	No
Model	20	Text	No
Cost	6	Numeric: currency	No
Sold	1	Character	No

Validation Routines

What are validation routines?

- Validation routines are **checks placed on data being entered** to ensure it matches the design of the system
- Validation routines **prevent errors** and **maintain data integrity**

Validation routine	Description	Examples
Range check	Ensures the data entered as a number **falls within a particular range**	For children aged between 5 and 10, checks to make sure numbers <5 and >10 are not accepted
Length check	Checks the **length of a string**	Password must be a minimum of 8 characters, checks to make sure the length is >=8

Type check	Check the **data type** of a field	Enter distance in whole miles, checks to make sure decimals are not accepted
Format check	Ensures that the data has been entered in the **correct format**	Enter date of birth (DD/MM/YY), check to make sure the format is correct
Presence check	Looks to see if **any data has been entered** in a field	Enter your username ,check to make sure the filed has not been left blank
Check digit	Check digits are **numerical values** that are the **final digit of a larger code**	Barcodes

Your notes

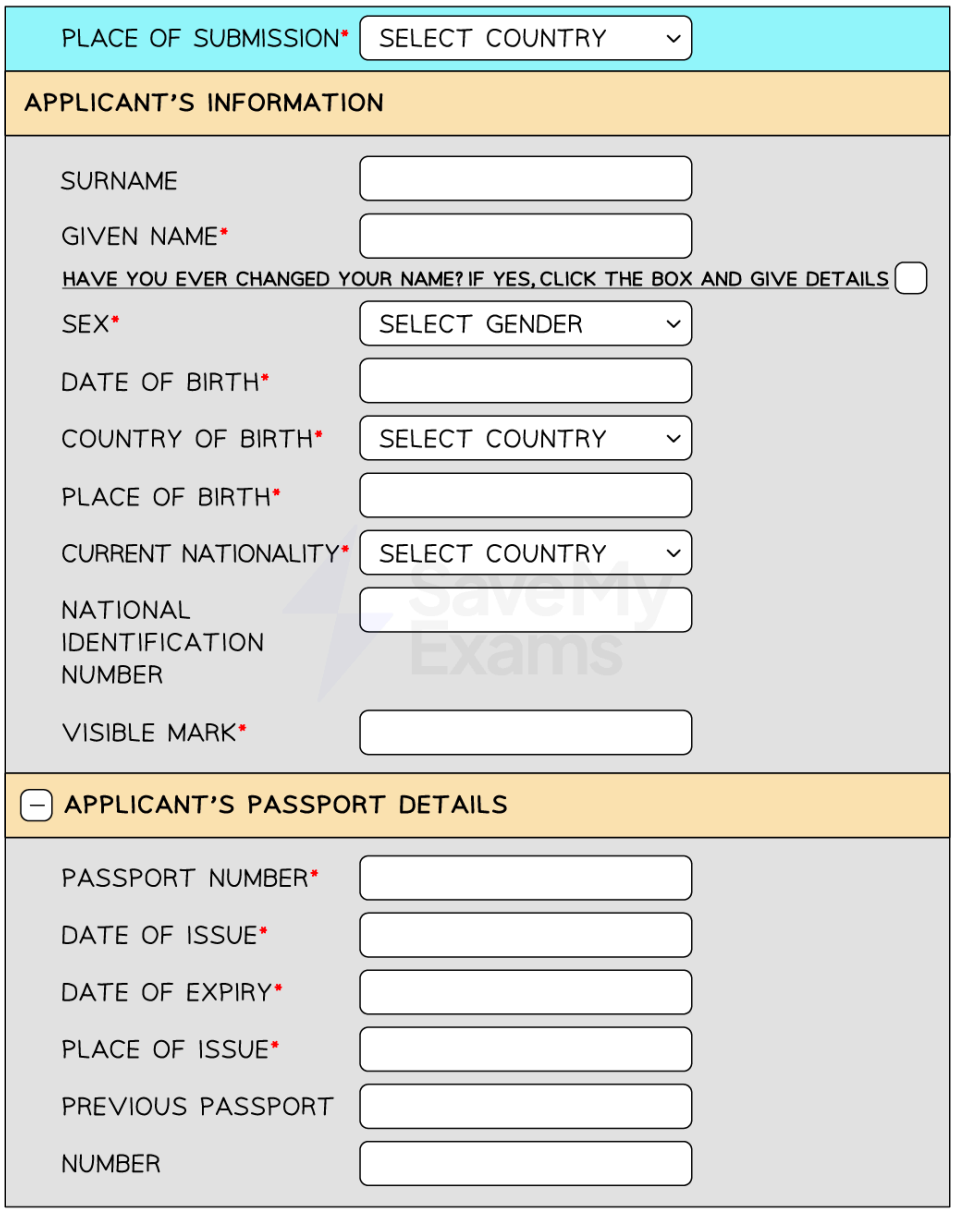
Input & Output Formats

What are input and output formats?

- Input and output formats are **design considerations** for **how data is captured** from (**input**) and displayed to (**output**) the users of a system

Input formats

- A data capture form is **used to collect data from a user in a structured format**
- Data capture forms should aid the collection of data by:
  - **Having a user-friendly layout** - short, visually appealing, logical flow
  - **Providing clear instructions** - concise instructions at the beginning
  - **Using appropriate question types** - multiple choice, drop downs, checkboxes etc.
  - **Using validation routines**



Your notes

Output formats

- Showing the results of processing either paper-based on on a screen should consider:
  - **Screen layouts**: how information is presented to users on a screen
  - **Report layouts**: how information is organised in a printed or digital report
  - **Readability**, **visual appeal**, and **efficient use of space**

Worked Example

Carlos is designing a new computer system to replace an existing system.

Tick four items which will need to be designed.

[4]

	Tick
Inputs to the current system.	
Data capture forms.	
Report layouts.	
Limitations of the system.	
Observation methods.	
Improvements to the system.	
User and information requirements.	
Validation routines.	
Problems with the current system.	
File structure.	

Answers

	Tick
Inputs to the current system.	
Data capture forms.	X
Report layouts.	X
Limitations of the system.	
Observation methods.	
Improvements to the system.	
User and information requirements.	
Validation routines.	X
Problems with the current system.	
File structure.	X

Your notes

Your notes

What is the system life cycle?

- The system life cycle is **a structured process** that guides the **planning**, **creation**, **testing**, and **deployment** of an information system
- Ensures systems are **systematically developed**, meeting needs of stakeholders, **minimising** risks and **maximising** efficiency
- The third stage is **testing**, the purpose is to:
  - **Verify** that the system **meets all requirements**, **functions correctly**, and is **free of bugs** through various testing methods.

Test Strategies

What are test strategies?

- A test strategy is **a method of testing a completed system** to ensure **all parts work as intended**
- Systems are designed in a modular format
- Each module needs to be **tested independently**
- Testing **is done again** once all **modules are joined together**
- The results of testing **may indicate changes** need to be made, **testing is repeated** once all changes have been made

Test designs

Test design	Why?
Data structures	Test all data is stored correctly
File structures	Test all data is stored in the correct format
Input formats	Test that data can be entered correctly
Output formats	Tests that screen output and reports are in the correct format
Validation routines	Tests that the system rejects unreasonable data being inputted

Test plans

- A test plan is designed and implemented **to ensure thorough testing of a system**, it includes:
  - **Test data**: specific data used for testing purposes

- **Expected outcomes**: predicted results based on test data
- **Actual outcomes**: results obtained from testing
- **Remedial action**: steps taken to fix identified issues

Your notes

Test Data

What is test data?

- Test data is specific data used for testing purposes
- There are **four** main categories of test data that would be used whilst performing final testing on a system
  - **Normal**
  - **Extreme**
  - **Abnormal**
  - **Live**

Normal data

- Normal test data is data that **should be accepted** in the system
- Normal data **has a known outcome**

Extreme data

- Extreme data is **data that is on the limits of what is acceptable**

Abnormal data

- Abnormal data is data **outside of the limits of what is acceptable**
- Abnormal data should be rejected by the system

Live data

- Live data is **data from the old system used on the new system**
- Live data **has known outcomes**, so it can be compared to the new system to ensure the new system works

Worked Example

The European Space Agency (ESA) is building a new space telescope to orbit the Earth and search for distant galaxies. The ESA is using computer controlled robots to build the lens of the telescope. A new computer system will operate the space telescope; the new computer system is made up of several modules.

Describe how the new computer system is to be tested before it is fully operational.

[4]

Answer

four of:

Each module has to be tested independently to ensure it functions correctly [1]
Modules need to be tested together [1]
Data needs to be transferred from module to module to check for data clashes [1]
Errors need to be noted and corrections made [1]
Then tested again [1]
The system as a whole needs to be fully tested under controlled conditions [1]

Your notes

Your notes

What is the system life cycle?

- The system life cycle is **a structured process** that guides the **planning**, **creation**, **testing**, and **deployment** of an information system
- Ensures systems are **systematically developed**, meeting needs of stakeholders, **minimising** risks and **maximising** efficiency
- The fourth stage is **implementation**, the purpose is to:
  - **Deploy** the system to the **production environment** and **ensure** all components work together in the **live setting**.

System Implementation

What is system implementation?

- System implementation is **a process** that happens **after a system has been fully tested** and is working correctly
- Implementation happens in **two stages**:
  - **Data is transferred from old to new system**
  - **System changeover**
- A changeover is **moving from the old system to the new system**
- A Changeover can occur in **four** different ways
  - **Direct**
  - **Parallel**
  - **Pilot running**
  - **Phased**

Direct changeover

- The **old system is replaced by the new system immediately**
- Used when quick implementation is necessary

Parallel running

- **Both old and new systems run simultaneously** for a period before the old system is phased out
- Used when a smooth transition with minimal risk is required

Pilot running

- The new system **is implemented in a small, controlled environment** before full-scale implementation

- Used when testing the new system in a real-world setting

Phased implementation

- The new system **is implemented in stages**, with each stage replacing a part of the old system
- Used when a gradual transition is preferred to minimise disruption

Implementation	Advantages	Disadvantages
Direct changeover	▪ Fast implementation ▪ Cost-effective as only one system is in operation	▪ High risk of failure ▪ No fallback ▪ Users can't be trained on the new system ▪ No backup of the system
Parallel running	▪ Lower risk ▪ Easy comparison of systems	▪ Time-consuming ▪ Resource-intensive
Pilot running	▪ Low risk as only trialled in one department/centre/branch ▪ Allows for fine-tuning ▪ Staff have time to train with the new system ▪ Few errors as it's fully tested	▪ Slower implementation ▪ Potential inconsistencies ▪ Confusion as there are 2 systems in use ▪ No backup for the department/centre/branch using the new system
Phased implementation	▪ Reduced risk ▪ Easier to manage	▪ Takes longer ▪ Potential compatibility issues

Worked Example

Tick (✓) the most appropriate method of implementation to match the statements below.

	Direct	Parallel	Pilot
All of the benefits are immediate.			
If the new system fails the whole of the old system is still operational.			

Your notes

This is the cheapest implementation method.			
The system is implemented in one branch of the company.			

[4]

Answers

	Direct	Parallel	Pilot
All of the benefits are immediate.	X		
If the new system fails the whole of the old system is still operational.		X	
This is the cheapest implementation method.	X		
The system is implemented in one branch of the company.			X

Your notes

Your notes

Technical Documentation

What is technical documentation?

- Technical documentation is **detailed information on the system's inner workings and programming** for developers and IT staff
- Technical documentation enables the system to be:
  - **Maintained**
  - **Repaired**
  - **Updated**

Technical documentation includes:	
Purpose of the system/program	Explanation of the system's intended function and goals
Limitations	Known constraints or issues with the system
Program listing	The code or scripts used in the system
Program language	The programming language used to develop the system
Program flowcharts/algorithms	Visual representations or descriptions of the system's logic and processes
System flowcharts	Visual representations of the interactions between system components
Hardware & software requirements	Necessary equipment and software to run the system
File structures	Organisation and layout of the system's files and data
List of variables	Collection of variables used within the system, including their names and purposes
Input format	Structure and format for entering data into the system
Output format	Structure and format for presenting data generated by the system

Sample runs/test runs	Examples of system operation, including input and expected output
Validation routines	Techniques used to check and confirm the accuracy of data entered into the system

Your notes

User Documentation

What is user documentation?

- User documentation is **instructions and guidance for end-users on how to operate the system**
- User documentation enables end-users to **effectively use the system** and **overcome problems**

User documentation includes:	
Purpose of the system	Explanation of the system's intended function and goals
Limitations	Known constraints or issues with the system
Hardware & software requirements	Necessary equipment and software to run the system
Loading/running/installing software	Instructions for setting up the system on user devices
Saving files	Procedures for storing data within the system
Printing data	Steps to produce hard copies of system data
Adding records	Instructions for creating new entries in the system
Deleting/editing records	Guidelines for modifying or removing existing entries in the system
Input format	Structure and format for entering data into the system
Output format	Structure and format for presenting data generated by the system

Sample runs	Examples of system operation, including input and expected output
Error messages	Explanations of system warnings and error notifications
Error handling	Steps to resolve issues and errors within the system
Troubleshooting guide/helpline	Assistance for diagnosing and addressing common problems
Frequently asked questions	Answers to common user inquiries
Glossary of terms	Definitions of key terms and concepts related to the system

Your notes

Worked Example

Following the implementation of the system, technical documentation needs to be written.

Identify three components of technical documentation which are not found in the user documentation.

[3]

Answers

three of:

program listing [1]
program language [1]
program flowcharts/algorithms [1]
system flowcharts [1]
file structures [1]
list of variables [1]
test runs [1]
validation routines [1]

Your notes

Evaluate the Solution

How do you evaluate a solution?

- To evaluate a system you must assess:
  - **The efficiency of the solution**
  - **The ease of use of the solution**
  - **The appropriateness of the solution**
- After successfully evaluating the system you will be able to **identify limitations and propose improvements**

Efficiency of the solution

- Evaluate the system's performance in terms of:
  - **Resource usage**
  - **Time**
  - **Cost**
- Consider whether the system is **operating optimally** or if improvements could be made to its **efficiency**
- Provide examples of specific aspects that contribute to the system's efficiency
- Identify areas that may be consuming excessive resources or time, and suggest ways to optimise them
- Questions to ask:
  - **Does it operate quicker than the previous system?**
  - **Does it operate by reducing staff time in using the system?**
  - **Does it operate by reducing staff costs?**

The ease of use

- Examine **how user-friendly and accessible the solution is** for its intended audience
- Assess whether the system is easy to learn and use, and if users can accomplish their tasks without difficulty
- Describe the user interface and how it facilitates interaction with the system
- Mention any feedback from users regarding their experience with the system, and address any issues they encountered
- Questions to ask:

- **Are all the users able to use the system and make bookings easily?**
- **Are all the users able to change and cancel bookings easily?**
- **Can all staff understand how to use the system with minimal training?**


Your notes

The appropriateness of the solution

- Compare the implemented solution with the original task requirements and **evaluate how well it meets the intended purpose**
- Outline the initial objectives of the system and discuss how the solution addresses each one
- Highlight any requirements that may not have been fully met and discuss possible reasons for this
- Collect users' responses to the results of testing the system
- Feedback can provide insights into potential issues and improvements, and help determine overall user satisfaction
- Summarise the testing process, including test data, expected outcomes, and actual outcomes
- Discuss users' reactions to the system, addressing any concerns or suggestions they may have
- Questions to ask:
  - **Is the system suitable for each of the departments?**
  - **Does it meet the needs of the customers?**
  - **Does it meet the needs of the staff?**
  - **Does the solution match the original requirements?**

Identify limitations and propose improvements

- Based on the analysis of efficiency, ease of use, appropriateness, and user feedback you should now be able to:
  - **List the limitations and provide explanations for each one**
  - **Recommend specific changes or enhancements that could address these limitations and improve the system**

Worked Example

Evaluation is a part of the systems life cycle. Describe two evaluation strategies.

[2]

Answer

Two from:

- Compare the final solution with the user requirements
- Identify any limitations of the new system
- Identify any further improvements to the new system
- Analyse feedback from users of the new system
- Compare test results from the new system with the old system

Your notes